I.T.E.C.

I.T.E.C. has been created to help everyone who would like to acquire a good basic knowledge of electronics and computers. This is the 'new technology' which over the next two decades will bring about an astonishing level of change — with an impact on our lives that most people find hard to imagine.

To be completely ignorant of the fundamental facts of electronics and computer science is at best hiding your head in the sand, and at worst may soon leave you feeling like a new kind of illiterate! As electronic devices have more and more influence on our lives it becomes increasingly important to know how they work, what they can do, and how to communicate with them.

I.T.E.C. is designed to teach these technical ideas to the interested non-specialist, and to offer a valuable aid to anyone studying these topics. It covers the whole field in detail, starting with basic principles and progressing to a more advanced understanding of both background theory and practical applications.

I.T.E.C. is divided into a number of major sections. The first half of the course concentrates on three main subject areas: **Solid State Electronics, Digital Electronics,** and **Computer Science.** These explain the real 'nuts and bolts' of how electronic systems work, and the design and use of computer software. Background pages on **Electrical Technology** provide a simple theory outline for those who have no knowledge of physics.

Further sections explain the design and use of **Microprocessors** and how electronics operate in the fast developing field of **Communications**. **Computers and Society** considers the key issues involved in the ever-growing range of computer applications.

The series is constructed as a *self-teaching course,* which builds your understanding step by step. The chapters are carefully planned to help you learn in progressive stages and each one moves you to a new level of knowledge. So to get the most from the course it is best to read each chapter before you move on to the next, rather than try to skip about. This means you will avoid meeting unfamiliar terms and ideas which have been explained in a previous chapter.

1

BASIC COMPUTER SCIENCE

Introducing the computer —past, present and future

Defining a computer and its modules

We live in the computer age. So many aspects of our daily lives are already touched in some way by computers – they sort our mail, handle our bank accounts, print out bills, run our transport systems, control industrial machinery and do a thousand-and-one other things.

Despite the speed and efficiency which computers can offer, some people are alarmed at how rapidly they are 'taking over'. But it's important to remember that, contrary to common belief, the computer can do nothing on its own – it needs people to make it work.

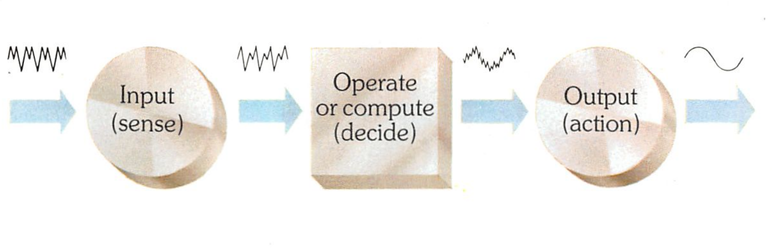
Computer science is concerned with information – its representation, processing, storage and transmission. This has become a vital function for most aspects of life in our complex modern society – be it

Tony Stone Photo Library – London

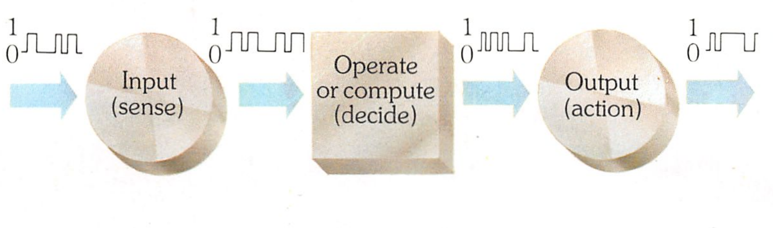
1. Computers are a familiar part of the scene in many areas of life today – from teaching aids (above) to the control of huge complex systems, such as the New York Stock Exchange (left).

industry, government, education, medicine, defence or whatever. Information means more than the daily news on the radio or television; it also means census records, currency exchange rates, library catalogues, the data needed to solve scientific problems or to control an industrial process. A new term, 'information technology', has come into common use, and is often used to describe the broad field of computer science.

2



3



The basic functions of an analogue computer (2) and a digital computer (3). Analogue devices work on information in the form of continuously varying signals; digital translate information into numerical codes based on just two values, 0 and 1.

What is a computer?

The computer is a machine that handles information. In the most general sense, as is shown in *figures 2 and 3*, it is a piece of equipment that accepts data at its input, processes the data according to a set sequence (calculations/decisions) and supplies the results in the form of data or actions.

This broad description of a computer could also apply to older types of calculator and data processing machine which are purely mechanical or electro-mechanical. (An electro-mechanical device has moving parts driven by electrical power, while an electronic device has effectively no moving parts). But in practice today the term computer is used almost exclusively to refer to an automatic digital electronic machine.

Some devices which operate on analogue rather than digital signals are also genuine computers. It's worthwhile understanding the difference between the two, although this series will be concentrating on the digital computer.

Analogue computers

This type of computer, as shown in *figure 2,* works on information in the form of continuously varying values which represent measurements of stresses, length, speed of rotation and so on.

An example of a simple mechanical analogue computer is the speedometer in a car. This receives a continuously varying signal from a device measuring the speed of rotation of the drive shaft and translates it into a movement of the speedometer indicator, to show the speed of the car.

An example of the use of an electronic analogue computer is the aeroplane flight simulator. The computer accepts analogue inputs generated by the pilot moving the controls, processes them and then generates analogue output, making the simulator move in response in a way that faithfully reproduces flight conditions.

Digital computers

There are three terms by which a computer today is usually defined: automatic, electronic and digital.

It is **automatic** in as much as, once it has been correctly set up and started, it functions by itself without any external intervention. It is **electronic** in that the core of its operation is based on electronic technology, although it does have mechanical and electromechanical parts. It is **digital** because it works with numbers (digit means single numbers between 0 and 9). These numbers, suitably coded, can represent either numerical data or data of any other type.

This type of computer, as shown in *figure 3,* works on discrete values which represent numbers – compare this with *figure 2* to see the difference to an analogue system. Almost all electronic computers use binary numbers or bits (from BInary digiTs) to represent information. A bit is the fundamental element of all data and can assume just two values: zero (0) or one (1).

A digital electronic computer can be schematically represented by three subsystems, or modules, as is shown in *figure 4.*

These are the input/output (I/O) unit, the memory and the central processing unit (CPU).

Input/output unit (I/O)
The I/O unit has two functions. The input unit provides the means for you to enter programs, commands and data. The output unit allows the CPU to return the answers to you or to another computer.

Memory
Here are stored the programs which tell the computer what to do. Also, data to be processed is read into the memory and held until ready for use, and the results of computations by the CPU are stored in memory until ready for output. This type of memory is called a central or working memory. If the memory space is not large enough it can be expanded with auxiliary or mass memory. These are normally in the form of discs, magnetic tape or drums physically separate from the central memory. They are used to hold large quantities of information such as program libraries or data bases for processing. The auxiliary memory and I/O devices are usually known as peripherals.

The computer's central memory is subdivided into elementary units called cells, locations or more commonly **words**. Each of these consists of a certain number of **bits**, that form a letter, symbol, etc. The number of elements can vary from computer to computer, but is fixed for each particular machine.

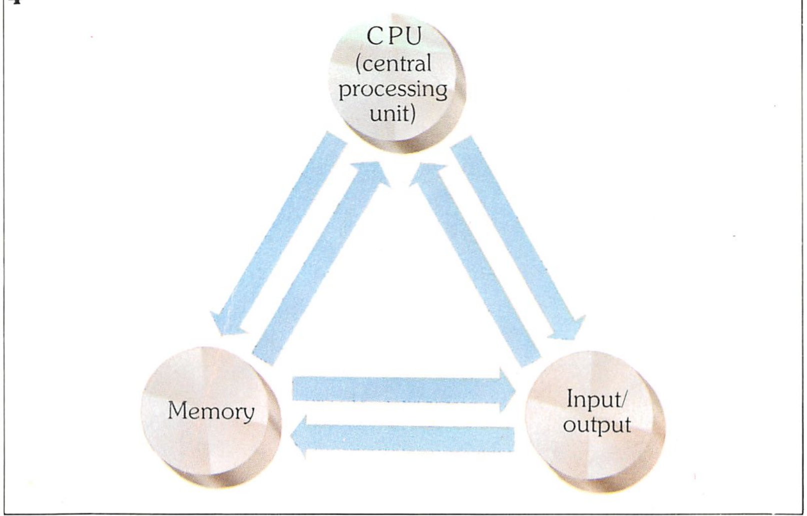
The number of bits in a word is called the word length. An 8-bit word is given the special name **byte**. The number of words in a memory is called the **memory capacity.** It's worth noting, however, that sometimes the memory capacity is expressed as a function of the total number of bits. The byte has in fact become the preferred word length with the advent of microprocessors.

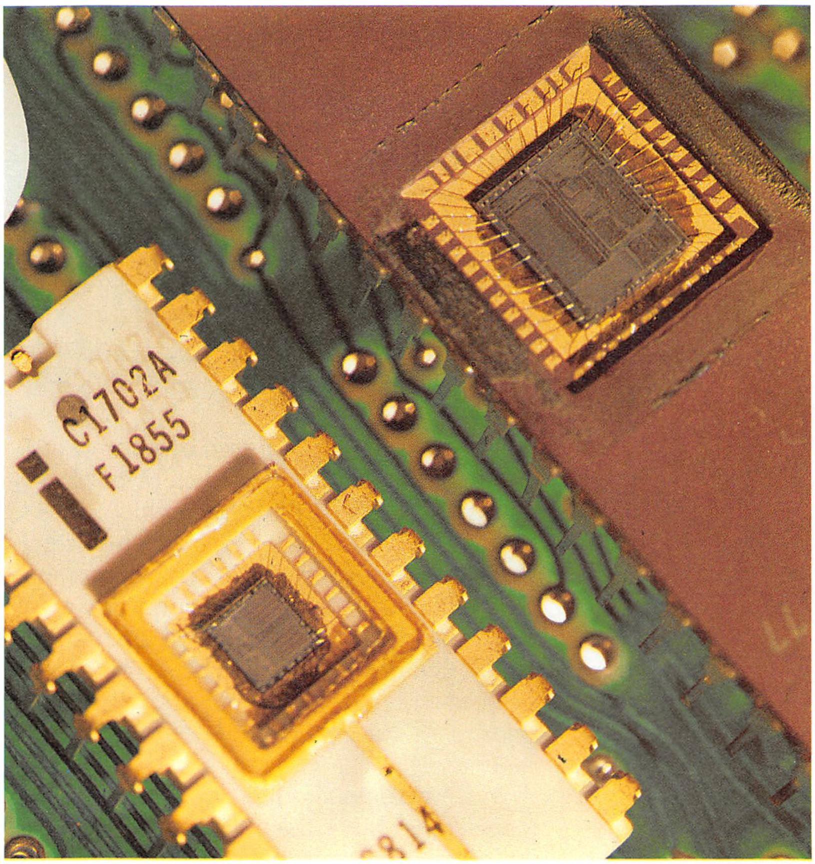
Central processing unit (CPU)
The CPU is the 'brain' of the computer. It reads the program in the memory, carries out calculations, transfers data, performs all the operations requested by the program and controls the overall functioning of the computer.

Two other terms we need to define are hardware and software. **Hardware** describes the physical elements of a computer, in other words those tangible parts that can be seen and touched. **Software** describes the intangible elements of a computer – its programs, – although today the actual program when listed on paper or recorded on tape or a disc is also called software.

4. Basic elements in a computer system.

4





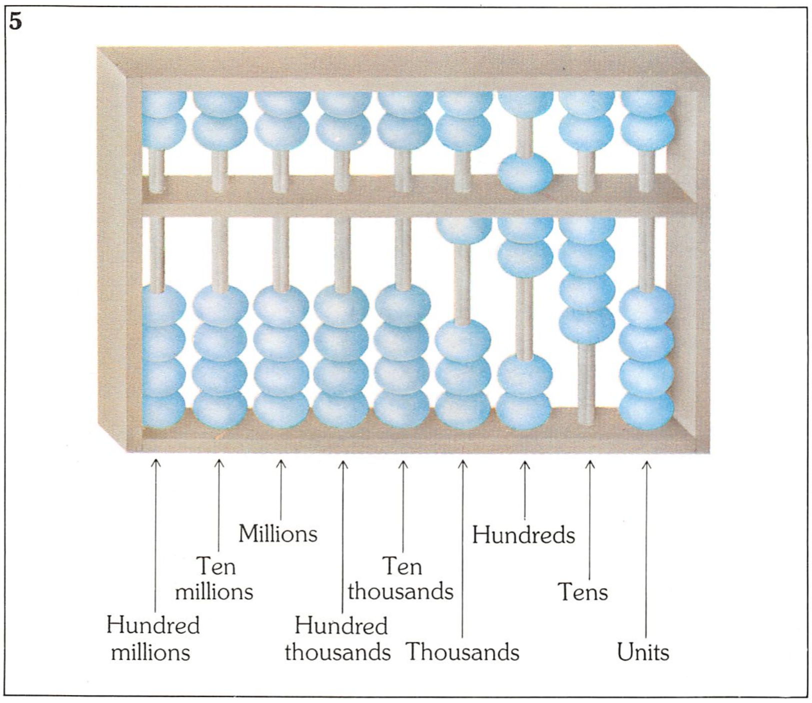
Paul Brierley

History of the computer

The evolution of the computer is a fascinating story, and tracing its step-by-step development gives a better understanding of some of the basic principles involved in today's computers.

You might say it began when Man found that counting on his fingers was no longer an adequate way to work out all the calculations he wanted to make. Early man probably started to calculate by tracing marks on the floor or walls of his shelter or by using groups of stones. These various systems were progressively refined until the first known calculator, the abacus, emerged.

5. An abacus – the number displayed is 1740.



A microprocessor CPU and memory.

The abacus, shown in *figure 5*, was invented around 2000 B.C. and is still widely used in many Asiatic countries. It consists of a number of vertical bars held in a rectangular frame and divided in two by a horizontal bar. Each vertical bar has a number of small movable balls on it. Each ball above the horizontal bar represents a group of five and each ball below the bar equals one. Reading from right to left the vertical bars represent units, tens, hundreds, thousands and so on. Zero is represented by all the balls positioned away from the central, horizontal bar. Numbers are entered by moving the relevant balls towards the horizontal bar starting from the right. The abacus shown in *figure 5* displays the number 1740.

Even though the abacus is an excellent and fast calculator (in the hands of an expert it can be faster, for some calculations, than a more modern mechanical calculator) it could not satisfy the ever growing need to reduce calculation time, the most boring part of any mathematical process. In 1617, John Napier, a Scottish mathematician, published an article which described the use of marked ivory sticks for multiplication and division. Napier's mathematical studies led to the invention of the slide rule.

In 1642 Blaise Pascal invented the first true adding machine which was similar to the mechanical table-top calculators still in use in the late 1960s. It was a complex system of wheels, gears and windows through which the numbers could be read. A similar, but more refined, machine was invented a little later by Gottfried Leibnitz. This machine could mechanically add, subtract, multiply, divide and even extract square roots.

Punched card systems

The next significant advancements came in the 1800s, when Joseph Jacquard produced a weaving machine, controlled automatically by punched cards. The idea of using punched cards to memorise instructions and data was later taken up by Charles Babbage who in 1835 invented a mechanical digital calculator, which was called an analytical engine. Babbage used punched cards to programme his machine, which was able to use the results of one calculation as the basis for another, without any need to re-configure the machine. Thus it was able to handle repetitive calculations.

Another important characteristic of the analytical machine was its ability to jump around in the program instructions rather than taking them in the entered sequence. In today's terminology this procedure is called conditioned loops and jumps, something which will be covered in detail in a later chapter.

The idea of using punched cards to hold information was a milestone in com-

puter development. For a long time the punched card represented the most widespread automatic method of storing and sorting data. The most widely used card type (see *figure 6*) was invented in 1890 by Hermann Hollerith. He played a major role in the development of data processing, and went on to found a company which was later to become International Business Machines (IBM).

He also invented the Hollerith code to represent data on the cards, and designed an electrical card-sorting machine. This was another crucial stage in the evolution of the computer since, until that time, only mechanical means had been used to handle the punched cards.

The Mark I computer

With the development of electricity, electromechanical machines began to dominate. In the electromechanical computer, relays and solenoids (electrically operated switches) were used. The first of these, called the Mark 1, was developed by Howard Aiken and others in 1944 at Harvard University.

It was a machine of enormous proportions, around 15.5 metres long and 2.5 metres high. Instructions were loaded by means of a punched paper tape and data was entered by means of punched cards. Results were recorded on cards by means of an electrical typewriter.

The Mark 1 computer could multiply two numbers in about 3 seconds. In 1947 the Mark 11 computer could carry out the same operation in less than a quarter of a second; that is to say 12 times faster. This was an enormous step forward at that time, but it still seems incredibly slow compared with today's computers which carry out hundreds of thousands of operations in one second. The development of electronics was the breakthrough which brought about this dramatic increase in speed, and the **thermionic valve** was a key factor.

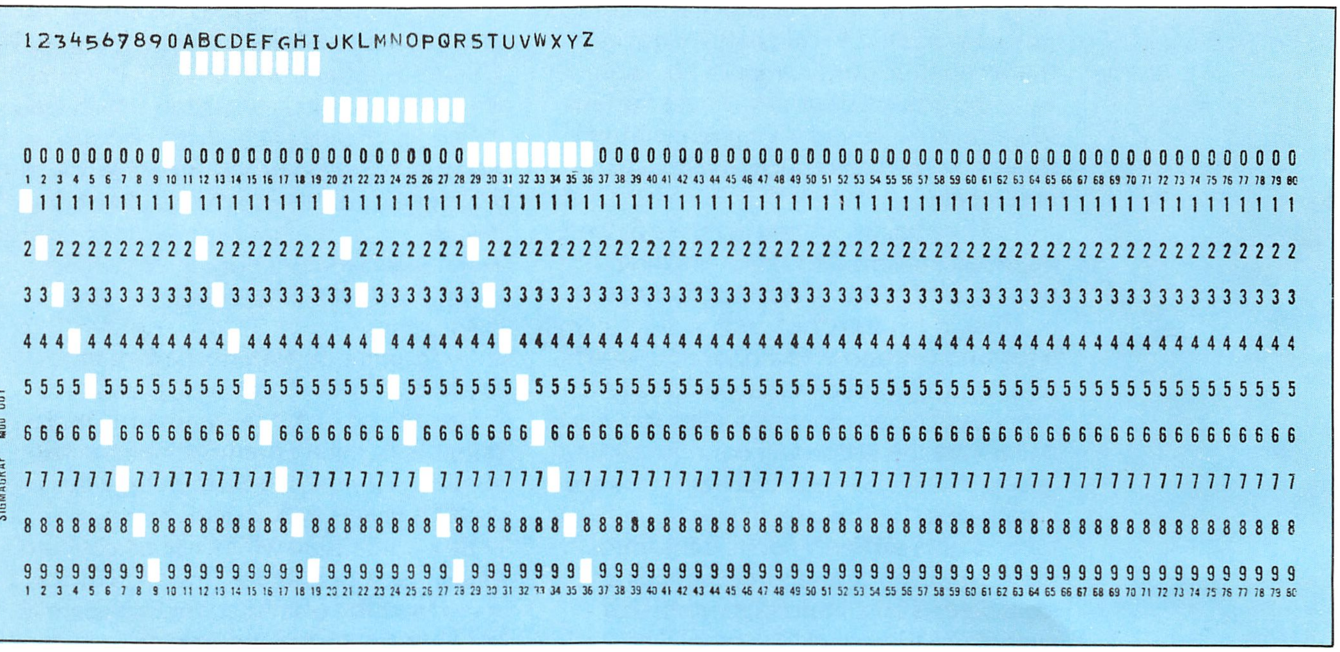
The first electronic computer

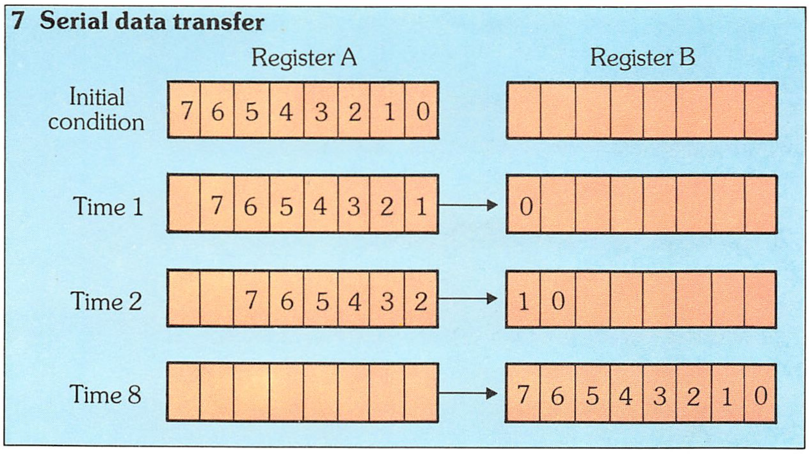
The valve is a device which for the first time permitted the electrical engineer to amplify electric signals and produce electronic memory cells.

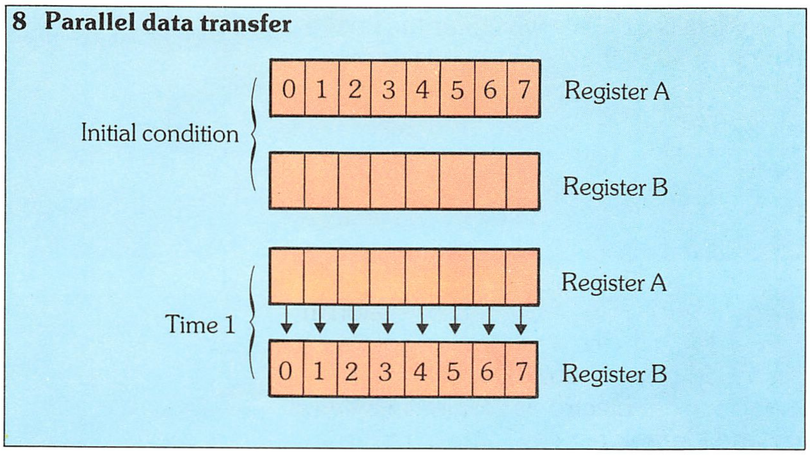
The first digital electronic calculator with valves was developed by G. Presper Eckert, John W. Mauchly and others from the Moore School of Engineering at the University of Pennsylvania in 1946. It was capable of carrying out a multiplication in about 2.8 milliseconds (1 millisecond = 1 thousandth of a second), and named ENIAC, from Electronic Numerical Integrator and Computer. The U.S. Army Ordnance Corps used the ENIAC to calculate aiming tables for the artillery. It was programmed by manually changing connections and setting switches which, as you can imagine, took a long time.

6. An IBM punched card.

6







7. Serial data transfer. The contents of register (memory cell) A are transferred to register B one bit at a time.

8. Parallel data transfer. The contents of register A are all transferred into register B in one operation.

During the same period, John Von Neumann of the Institute for Advanced Studies at Princeton New Jersey, published an article which had enormous importance in the evolution of the computer. One of the concepts presented was that of **parallel bit transfer.** In a serial machine the data bits are exchanged with the outside world and with the computer, one after another, as shown in *figure 7*. In a parallel machine all the data bits are transferred together at a given signal as in *figure 8*, like the start of a horse race.

Clearly a lot of time can be saved moving eight bits in a second rather than eight in eight seconds, and the basic principle is still used today, although in a real machine the exchange rate is much higher than one every second.

Another of Von Neumann's ideas has had even more impact on the construction of computers. This was the concept of holding the computer program in the computer's own internal memory. Developed and perfected, it is still used today.

Having the program in memory allows the computer to work much faster and gives it greater flexibility. This means it can switch from carrying out a scientific calculation to sorting a list of names or altering the quantities in a stores inventory system and then return again to the scientific calculation. The concept of the stored program has been the most significant element in making computers a commercial success.

Although new ideas in computer construction were continuously being perfected, the computer industry itself did not move from the sidelines until 1951 when the same group of people who had developed ENIAC produced the UNIVAC in marketable quantities.

This represents the starting point in what is known as the 'generations' of computer. Improvements in electronic circuits, in particular integrated circuits, together with the development of more efficient programming methods, have resulted in computers that are faster, more reliable, smaller, cheaper and use less power. This made it practical for computers to be used in all kinds of fields, beyond industry and big business.

For example, the Texas TI-59 programmable calculator, as shown in *figure 9*, which is now mass produced, runs off batteries, costs very little and yet has the calculating power of a large computer from the 1950s. The progress made in computing over the past 30 years has been quite astounding, and what's more we may not even be half way along the evolutionary path of the computer. If, up to now, the computer has had a marked effect on our lives, in the future it promises to have even more impact.

Computer 'generations'

The so-called generations represent the major stages in development of the modern digital electronic computer. The first generation began in 1951 with the UNIVAC 1, the first industrial computer, built by Remington Rand. The first UNIVAC 1 was installed in the Censors Office in the United States. The first IBM machine, the 701, was installed in 1953. A year later the 650 was introduced onto the market, and for many years it remained the

most popular computer available.

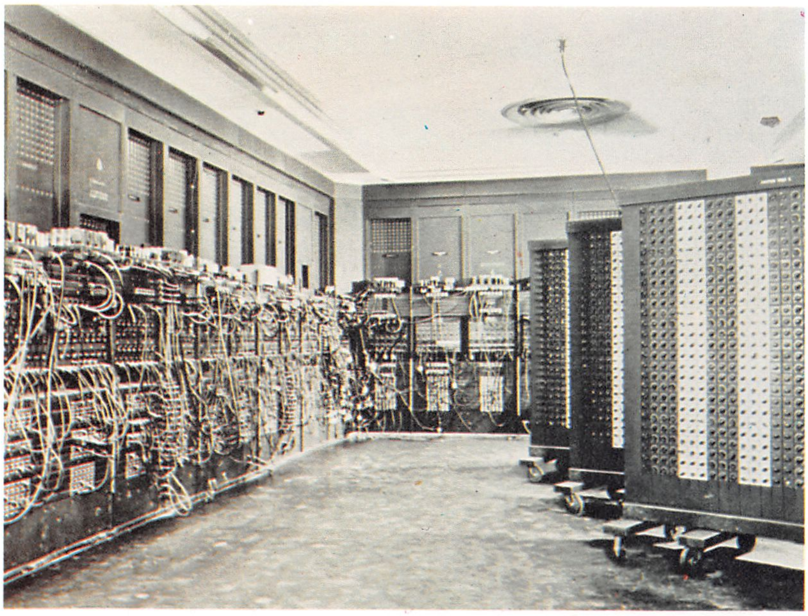
From a hardware point of view the first generation machines were constructed with valves. These absorbed a great deal of energy, generating so much heat that many forms of air conditioning were required for cooling. These units in turn also consumed vast amounts of energy. The reliability of the machines was so low that 'down time' often exceeded operating time. (**Down time** is the term used to describe the time that a machine is not capable of performing its job). Their memory capacity was quite limited and their speed of operation was in the order of two milliseconds (0.002 seconds) for a multiplication.

The second generation began in 1959 when **transistors** replaced valves. Transistors were more economical, more reliable and smaller than miniature valves; they also consumed less energy and produced less heat. As a result, circuit density could be notably increased – that is components could be placed closer to each other, so reducing the overall dimensions. Generally the second generation machines were smaller, more reliable and cheaper than the first generation and, for equal computing power, consumed much less electrical energy.

During the period of the second generation, many companies entered the field of computer science, designing and constructing computers. Some aimed at the general purpose market and others concentrated on scientific applications, giving high precision in mathematical computation. Table 1 shows some of the more widely used second generation machines. RCA specialised in management systems, IBM and UNIVAC in management and scientific systems, while Control Data concentrated on scientific applications and high speed computers.

The third generation – upward compatibility

With the second generation of computers came a problem: programs written on the smaller computers would not run on bigger machines, even if they came from the same manufacturer. This was the principal reason for the success of the computer family concept in the third generation of computers which were first produced in 1966. A company could begin by buying or renting a small machine and, as its needs increased, move on to a bigger machine from the same family. Every model was given a number – the more powerful the machine, the larger the number. Programs written for the small machines could be run on the bigger machines with little or no modification. This characteristic was called **upward com-**



9. Programmable pocket calculator. This has the same computing power as the type of massive machine of the 1950s pictured below.

10. The world's first electronic calculator. (photo kindly supplied by Sperry Univac)

Table 1
The most common second generation computers

Manufacturer	Model	Size	Applications
IBM	1620	Small	Scientific
IBM	1401	Small & medium	Business
IBM	7094	Large	Scientific and business
CDC	1604	Medium large	Scientific
CDC	3800	Large	Scientific
RCA	501	Medium	Business
UNIVAC	1108	Large	Scientific and business

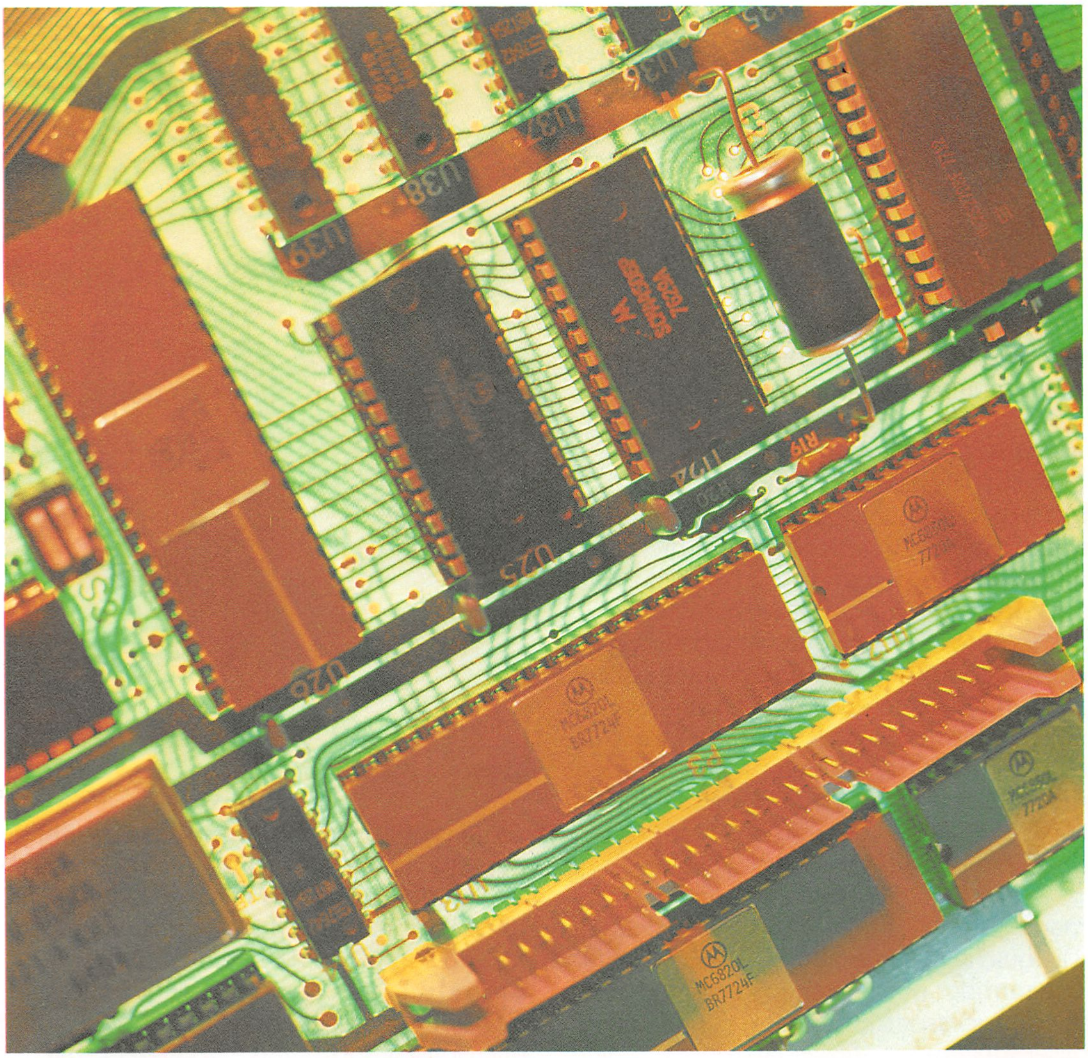
patibility. A famous example of this is the IBM 360 system, whose family progresses upwards from a small office machine through to large scientific systems.

Another characteristic of third generation machines was the use of miniaturised modules and integrated circuits. The miniaturised hybrid circuits (called by IBM 'solid logic') used transistors and capacitors mounted on a ceramic base (substrate). Resistors and conductors were photographically printed on to the substrate. The dimensions of the modules were around 127 mm square. Very soon however most manufacturers passed to the use of integrated circuits (IC).

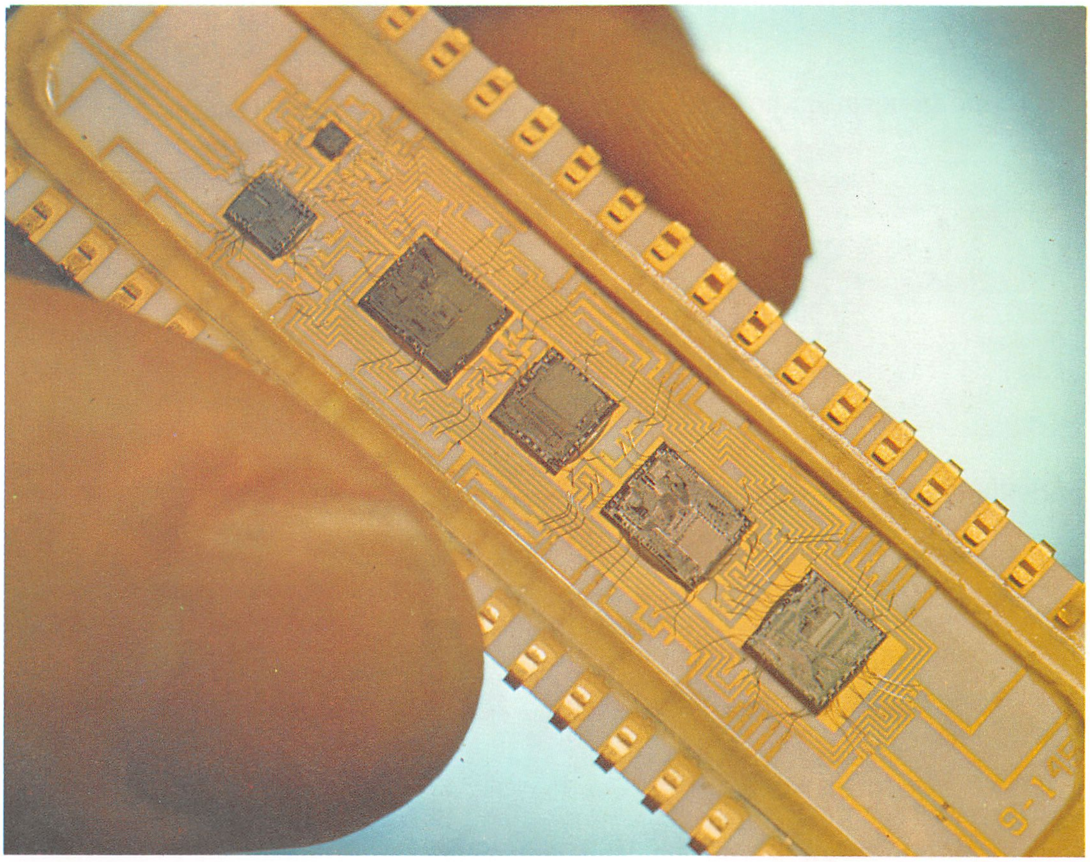
Since their invention in 1958, a vast range of different ICs have been designed and produced. A typical IC is a small package around 25.4 mm long, 12.7 mm wide and 3.4 mm thick. Inside the package is a small silicon chip that contains a large number of interconnected transistors and other components as shown in *figure 11*.

The first integrated circuits contained only a few components but, as the technology became refined, more and more

Section of a single-board micro-computer showing a range of integrated circuits mounted on a printed circuit board.



Paul Brierley



Paul Brierley

A number of integrated units are mounted on a substrate to form a more complex hybrid unit. This type of technology was rapidly overtaken by progress made in circuit integration.

The microprocessor (not to be confused with the term 'micro-computer') is a vital operating component in modern computers. The table below lists some of the most widely used.

components were packed in. Third generation computers which use integrated circuits as the fundamental component are smaller, more reliable and cheaper than second generation machines, and for equal computing power consume less energy.

The microprocessor

The difference between the third and fourth generations is not so clear as between preceding generations. Some say it began with the production of the IBM system 370 (an updated version of the 360), others that it began with the advent of the microprocessor. A microprocessor is a complete Central Processing Unit contained within a single IC. Further progress has now meant that other functions such as memory and I/O can be integrated on the same chip as the microprocessor, to produce in effect a single chip computer.

The first microprocessor on the market in 1971 had a 4-bit CPU which was quickly followed by an 8-bit CPU. The number of bits is an indication of a microprocessor's computing power; all other functions being equal, more bits give more power. The first microprocessor had, in these terms, rather limited performance.

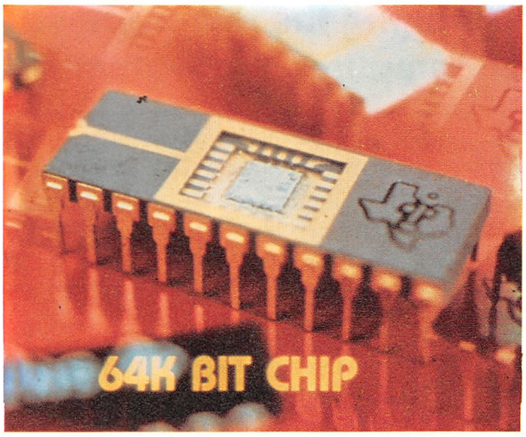
From the start of the 1970s, however, the microprocessor has become as powerful and sophisticated as the CPUs in second and third generation computers. Today there are also 16-, 32- and 64-bit microprocessors being marketed. Table 2

Table 2

The most common microprocessors

Bit size	Manufacturer	Type/number
4	Texas Instruments	1000
4	Intel	4004
8	Intel	8080, 8085
8	Fairchild	F-8
8	Motorola	6800, 6809
8	National	IMP-8
8	RCA	Cosmac
8	Mos Technology	6502
8	Signetics	2650
8	Zilog	Z-80
16	Texas Instruments	9900
16	Motorola	68000
16	Intel	8086
16	Zilog	Z-8000

An integrated circuit packaged with its terminals magnified approximately × 2.



11. A complete microcomputer on a single chip, magnified approximately ×20.
1 Read only memory (ROM)
2 Registers
3 Scan generator
4 Arithmetic logic unit
5 Clock
6 Segment decoder

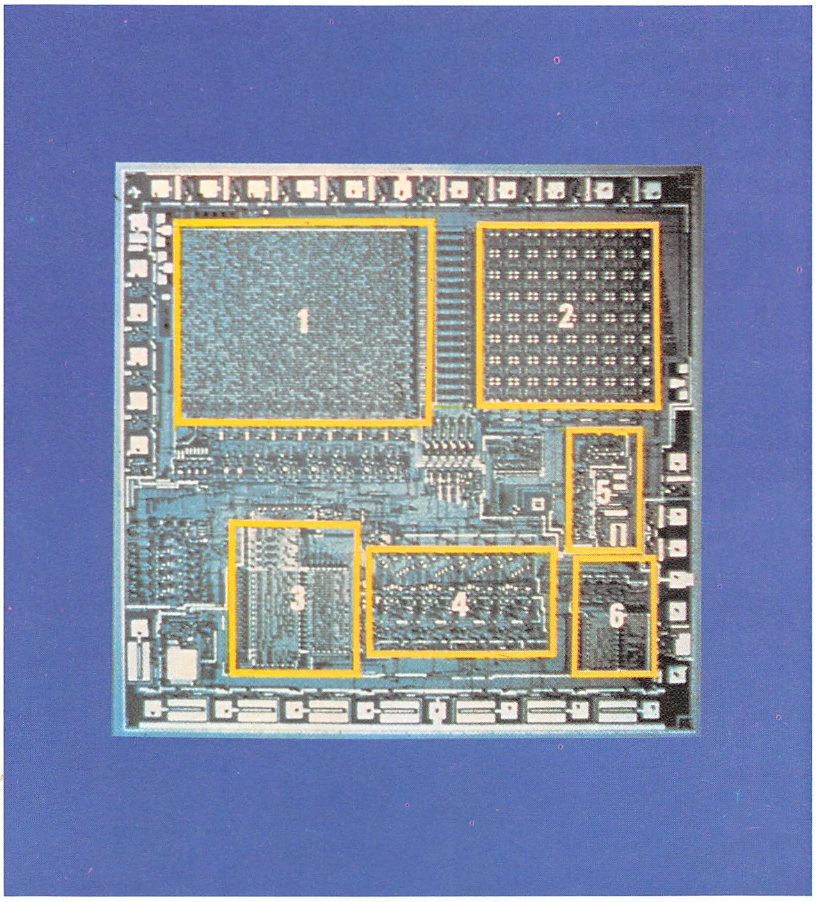


Table 3

Level of integration

Level of integration	Gates per chip	Year
Small scale of integration (SSI)	12	early 1960s
Medium scale of integration (MSI)	100	late 1960s
Large scale of integration (LSI)	1000	early 1970s
Very large scale of integration (VLSI)	50000	late 1970s
Very very large scale of integration (VVLSI)	greater than 100,000	early 1980s

lists some of the most widely used microprocessors. Examples of well-known fourth generation computers are the Apple II, Sinclair Spectrum and others like them.

Now a fifth generation of computers is under development in Japan. They will have the capacity to process a variety of data simultaneously, so many tasks can be carried out at once rather than one after another as in present computers. In addition, the computer has the capacity to adapt its own programs in the light of the information which is fed to it.

Even now, for example, computers used for medical diagnosis and geological surveying contain more information than can usually be remembered and recalled by a doctor or geologist. The data that computers can contain, their ability to recall it and to check cross references enables them to make deductions about the causes of a set of medical symptoms, say, or the possible presence of minerals within geological structures, which may not occur even to the expert.

Some see this as the beginnings of 'artificial intelligence' and the 'thinking' computer, in that, although the computer cannot originate thought, it can use the programs which have been fed into it to make deductions and new connections.

Improved integration levels

Microprocessors and single chip microcomputers are the result of refinements in integrated circuit production techniques, along with the development of techniques to increase circuit density. Circuit density is an indication of how many components and interconnections can be integrated onto a single chip. The maximum density is limited by production techniques, since the spacing is fantastically small, by the type of circuit and by how much electrical power it must handle.

Another term used to indicate this is **level of integration** which is measured by the number of gates on a single chip (a gate is the simplest logic element). Table 3 shows that the growth in the level of integration has increased 4000 fold over the past 20 years. At the same time, the area occupied by a single gate has been reduced and therefore the chip has not

increased proportionally. In reality chip dimensions have only increased by a factor of 25 since the sixties.

In the meantime production processes have been improved and have become less costly, to such an extent that the cost of a single gate has been reduced by a factor of 1000 and the overall chip costs have not increased significantly. This has provided very low cost CPUs with outstanding computing power, for use in computers that are small, use low power and are highly reliable.

The high level of circuit integration has had a significant impact on the evolution of computers; in particular Very Large Scale Integration (VLSI) has given rise to large capacity semiconductor memories, in addition to low cost microprocessors and microcomputers. Memories of 65536 bits, are currently in use. This is generally referred to as a 64 K memory. Memory capacity is normally given in bits, K-bytes or K-words, the K signifying 10 24 or 2^{10}, much as k is used to signify 1000 in other systems of measurement.

This extraordinary evolution of the hardware, together with improvements in software and programming techniques promises a continued rapid evolution towards ever smaller, more powerful and efficient systems.

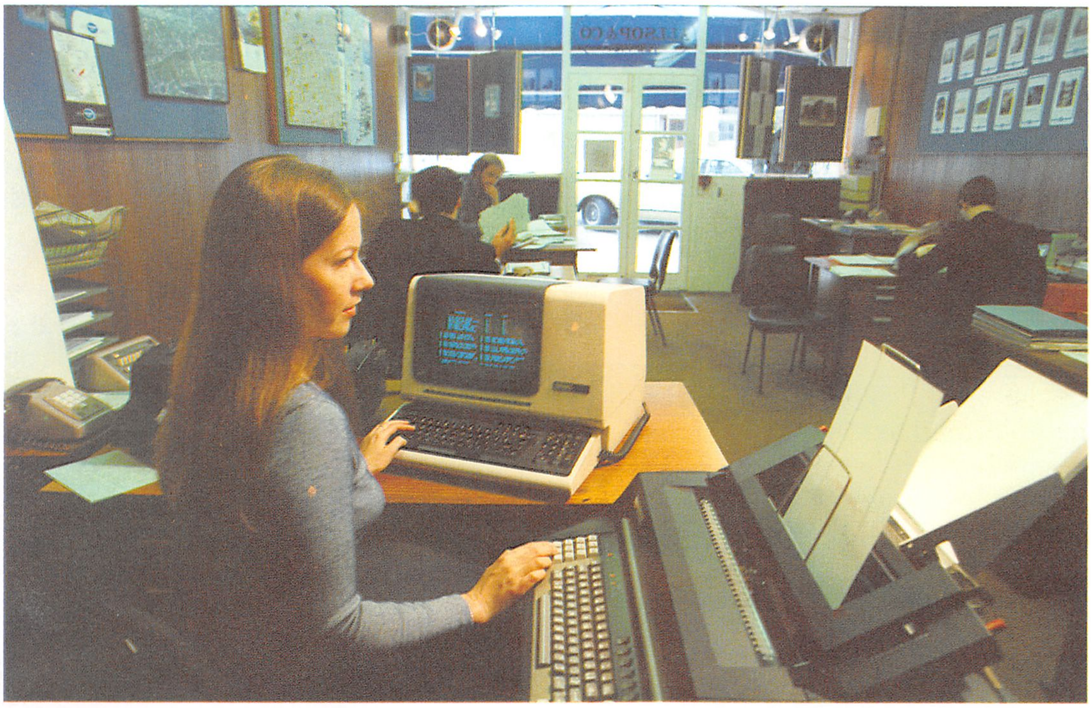
Outlining the development of computer programming

The computer, that is the hardware, is useless without a program. In some computers, especially those based on microprocessors, the program is integrated into the hardware when it is being made, written onto a Read Only Memory (ROM). As the name suggests these memories can only be read from and not written onto, and therefore the contents cannot be changed.

Some pre-programmed software in ROM is contained in special modules, as shown in *figure 12*, ready to be inserted into the system. Software contained in this type of permanent plug-in storage is sometimes called **firmware.** This gives very fast access to frequently used programs.

In general, however, the program is read (loaded) into the computer's memory from external devices such as paper tape, magnetic tape or discs. When the program has finished running the computer memory can be cleared and another program loaded ready to carry out other work. This is known as the Von Neumann's stored program concept.

Here we should take a quick look at early programming techniques. The first



Remote terminals linked to a central computer can give a large number of operators access to centrally-held information. This one is in use in an estate agents' office.

12. A pre-programmed software cartridge being inserted into a home computer.

commercially produced computers were used for such things as sorting and tabulation of large quantities of data, solving complex mathematical or scientific problems or carrying out long and repetitive processing. Few people truly knew how to use a computer, and those few who did had little idea of the computer's true potential.

The machine is programmed by telling the CPU what to do. In the early days, the machine was programmed in a particular way to suit each application, so that for every application a completely new program had to be written from scratch. Eventually it was realized that many of these individual programs contained parts common, or very similar, to other programs. So ways were found to group these common factors together into general purpose programs and procedures.

Computer programs must be written so that they can be understood by both man and machine. Humans can communicate with each other in all sorts of ways, from the simplest universal sign language to complex written or spoken languages.

The language of computers

Computers, like people, can understand many different languages – all they need is to be taught.

Machine language, made up of binary numbers, is the most straightforward and most universal. Remember that a binary digit can assume only the values 0 and 1, and that it is called a bit. Machine language is used for inter-machine communications. If men/machine communications had to be carried out in this way they would be almost impossibly difficult and extremely time consuming to read or write.

However, symbolic languages do exist, and these allow programs to be written using ordinary words, usually English, and decimal numbers. The simplest of the symbolic languages is 'Assembler'. This uses instruction abbreviations called mnemonics which are converted into machine language programs.

In so-called high-level symbolic languages, many machine code instructions are represented by a single symbolic instruction. This symbolic instruction often takes the form of a word, or words, used in common languages, so they are easily understood by people. Table 4 lists some of the high-level symbolic languages. FORTRAN (from FORmula TRANslation), for scientific applications, and COBOL (from Common Business Orientated Language), for business application, are two of the languages currently in widespread use. They have undergone many refinements since they were first formulated, and have become very easy to use.

To these can be added PL/1 (Programming Language 1), which was developed by IBM because of the need to generate compatible software. They took the best features of FORTRAN and COBOL and merged them in a single language. (Languages will be covered in more detail in a separate section).

Remember what we said about the upward compatibility of the IBM system 360. This concept can be applied to software in general, making it more easily transportable between one machine and another, easier to maintain and more efficient.

PASCAL was developed as a language to aid good programming techniques. We have already mentioned how programmers discovered gradually that common parts of programs for different applications could be grouped together and used in many applications. A result of

this continuing evolution in programming techniques has been the birth of **structured programming,** which makes use of various pre-existing concepts. One is top-down development of a program: this is the division of a program into different levels, every level being more detailed than the one preceding it. Another is the organisation of a program into functions. 'IF-THEN-ELSE', 'DO-WHILE', 'DO-TEST' structures are some examples.

Following these programming principles, a segment of a program written for a certain application can be used in another program for a different application. e.g. the comparison part of a program used to identify fingerprints may also be used in a system which is used for checking signatures. The savings in programming time made in this way can speed up the development of new systems and can also reduce their overall costs by a considerable margin.

BASIC (from Beginners All-Purpose Symbolic Instruction Code) is a simple but efficient programming language which can be used by people with little experience. BASIC instructions are very similar to normal mathematical operations and simple English language phrases. It is an **interactive** language, meaning it allows direct conversation between the machine and the user which greatly simplifies the work of programming. These characteristics make BASIC ideally suited to use in personal home computers which are now becoming so widespread.

Developments in software

The increasing use of microcomputers has accentuated the need for software that is **compatible**, in other words that can be used on machines that are different but able to understand the same language. Once someone has written a program for, say, a pay roll or file management it can subsequently be sold for use on compatible machines.

The term **software engineering** has been coined to describe the process of subdividing programs into smaller sections to make it easier to adapt them for use in other applications.

Although high-level languages make programs easier to write, they do in turn require a fairly complex program to translate them into machine code. Computer manufacturers normally provide a **compiler** built into the hardware, which in effect is a program which carries out this translation.

The arrival of the personal computer, and the corresponding increase in the number of inexperienced programmers, has created an urgent need for packaged programs supplied directly from the software manufacturer, such as word processing software, accounting software, mailing lists etc. The software industry is evolving in parallel with hardware to try and ensure that computers can be used easily and efficiently by a growing number of non-specialists.

Table 4

High-level program languages

Name	Characteristics
FORTRAN	Scientific use
COBOL	Business use
PL/1	Scientific and business use
PASCAL	Language for structural programming
BASIC	Ideal for home computers

Magnetic tapes (shown here in racks of spools) are used for the mass storage of information in large computer systems.

Tony Stone Photo Library – London

Glossary

binary numbers	method of counting in which all numbers can be expressed using only the digits 0 and 1
bit	abbreviation for binary digit. The smallest possible unit of information expressed in permutations of 1 and 0
byte	a word containing 8 bits
CPU	central processing unit of a computer, the 'brain' of the computer
firmware	permanent plug-in storage units containing computer software programs
hardware	the computer itself, and the actual machinery connected with it
input unit (I)	component used to feed information into the computer memory; eg. keyboard, paddle, magnetic tape, light pen
machine language	the binary language which the computer uses for its own internal processes
memory	the section of the computer where the information and program are held until needed. It can be divided into central memory and auxiliary memory
micro-processor	a complete CPU contained within a single integrated circuit
output unit (O)	section of computer from which information or results emerge in a useable form, eg. print out or video
peripherals	those parts of a computer which pass data to and from the CPU and which store information
program	pre-determined sequence of instructions
ROM (read only memory)	some microprocessor-based computers have the program already integrated into the system. This program is written onto a Read Only Memory component. The information in the program can be read from but not adapted and is therefore permanent
routine	sequence of programming instructions
software	term used for the actual program fed into a computer. Also used for programs listed on paper or recorded on tapes or discs
symbolic languages	special languages used by programmers to communicate with the computer
word	a certain number of bits which together form a letter or symbol

How electricity powers a system

Basic organization of electrical systems

Even the most sophisticated electronic systems rely on the use of electricity. So in studying electronics it's important to start by understanding the basics of electrical systems, and the broad facts of how electricity itself 'works'.

All electrical systems, however complex, either perform work or manipulate information – or do both of these things. This is part of a pattern known as the **Universal System Organization.** Any system can be broken down into three basic elements of organization – sense, decide and act.

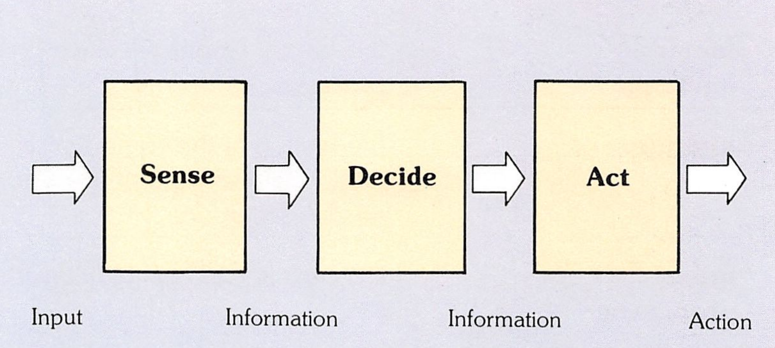
Figure 1 shows a block diagram of the Universal System. As you will see, all systems function in the same way: information enters, is handled internally and as a result some external action is carried out. Sometimes these fundamental stages of a system are referred to as 'input, process and output', or 'input interface' and 'output interface'.

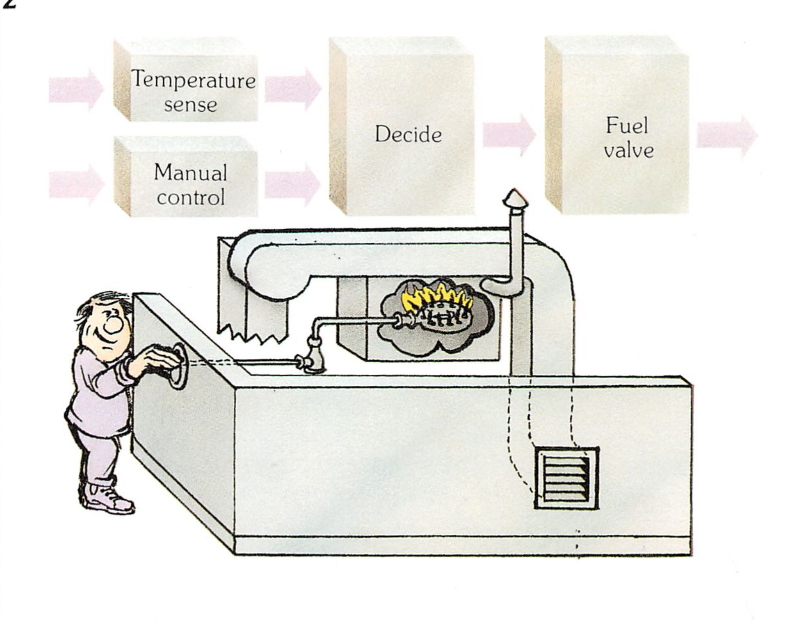
An example of the principle at work is shown in *figure 2*, which represents a functional or block diagram of a typical central heating system's thermostatic control. This system must have a temperature sensing device, which tells the system when the surrounding temperature has gone above or below the desired level. The control device tells the system what the desired level is. So these devices convert external input into internal information which can be dealt with by the system.

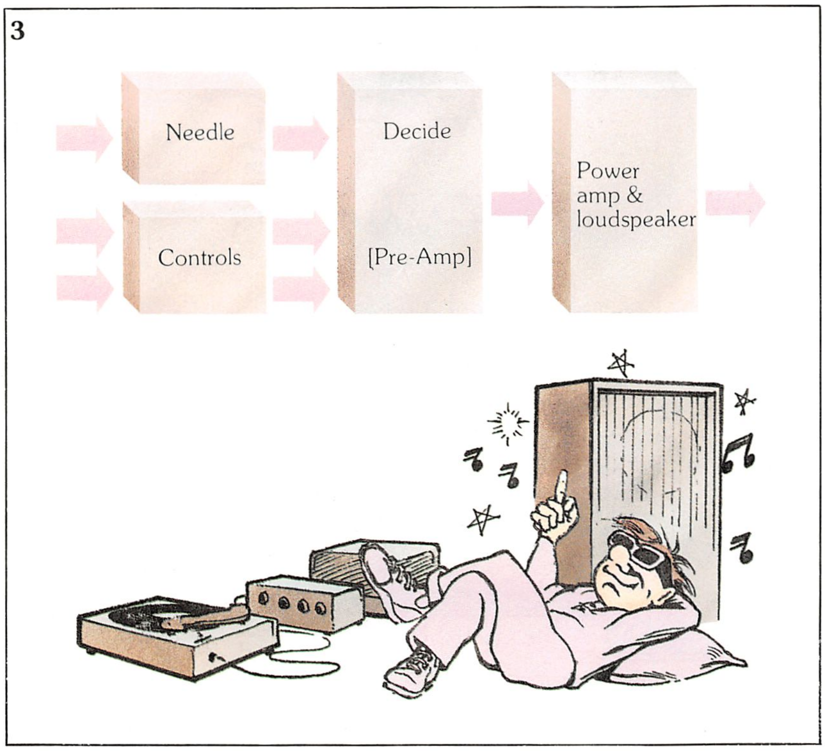
The system must then use these two streams of information to reach a decision. If the decision is to open the fuel supply valve, the valve mechanism converts this information into the action of moving the valve. So it's clear that the system is organized according to the Universal scheme: sense, decide, act.

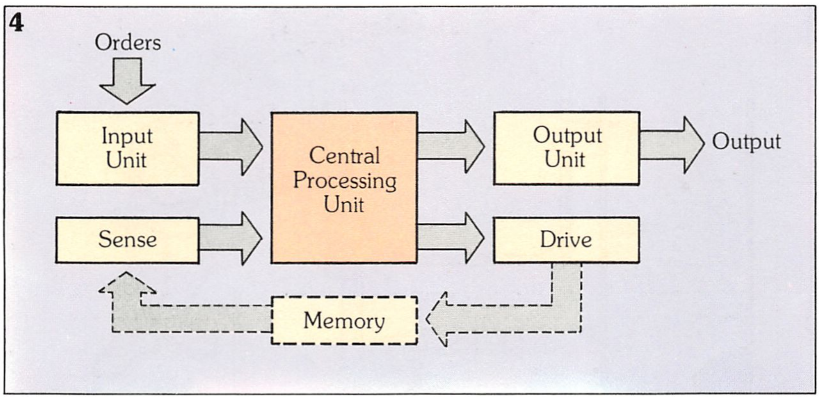
Figure 3 represents the block diagram of a hi-fi system, which operates following the same basic scheme. There is an input signal from the needle and the pickup (these components *sense* and transform the information contained in the groove of the record into electrical information). Information is also fed in from the manual

1. **Block diagram of the Universal System.**

1


2






2. **Block diagram** of a thermostatic control system for a central heating unit.

3. **Block diagram of a hi-fi system.**

4. **Block diagram of a computer,** showing how it is organized.

volume and tone controls. The input from these devices usually goes to a preamplifier which *decides* what the speaker ought to do, by setting the level of information entering the power amplifier. The amplified signal is passed to the loudspeaker which finally *acts* and produces the sound.

Figure 4 shows that a computer is also divided into three typical segments. In this case, however, the system is more complex, in that the sense or 'input' block is divided into two sections which gather two streams of information. Similarly, the action or 'output' block is also divided into two parts. The 'decide' segment is the central processing unit of the computer.

How systems use electricity

This is how electric and electronic systems in general are organized. But *how* do the systems manipulate information and do work? What property of electricity enables it to do these things – the voltage, the current or something else?

Electricity is actually fairly simple to understand because it behaves like a liquid. It flows like water and like water tends to fill every available space. Electricity is composed of minute particles called electrons which exist in every kind of matter. In metal wire, electrons can be pumped like water by a generator or by a battery. Electrons repel each other so they tend to reach the same density in all points of a circuit. It's rather similar to water seeking the same level under the influence of the force of gravity.

As there are many fundamental similarities between the behaviour of water and that of electricity, the properties of electricity can be illustrated quite well by using the example of a stream of water.

Figure 5 shows two tanks containing water, connected by a pipe. The man working away at the pump is the equivalent of an electricity generator. As the man pumps water from one tank to the other the water level, and so the pressure, rises in one tank and falls in the other. As a result the water is pushed through the pipe and flows from the high to the low pressure level.

Exactly the same thing happens in an electricity circuit. There is a source of electron pressure called a **generator.** The pressure on the electrons is measured in **volts** and may be produced by a generator, battery etc. These perform the same function as the man at the pump. The flow of electrons, like the water, is from a high pressure to a lower pressure area, only this time it is through a piece of wire. In the water system the flow rate is measured in litres/second, in an electrical system it is in Amperes (**Amps**). The flow of electrons is called a **current.** The relationship between pressure and flow is also very important.

In *figure 6* you can see that by

working harder the man has raised the level of the water in one of the tanks, so the pressure has been increased and more water flows through the pipe. In an electrical circuit, the same thing applies. The greater the electron pressure (voltage), the greater the flow of electrons.

The resistance factor
Yet another factor can influence both the flow of water and the flow of electricity. This factor is called **resistance**. In the case of water, the resistance is mainly created by how much the flow is constricted by the walls of the pipe.

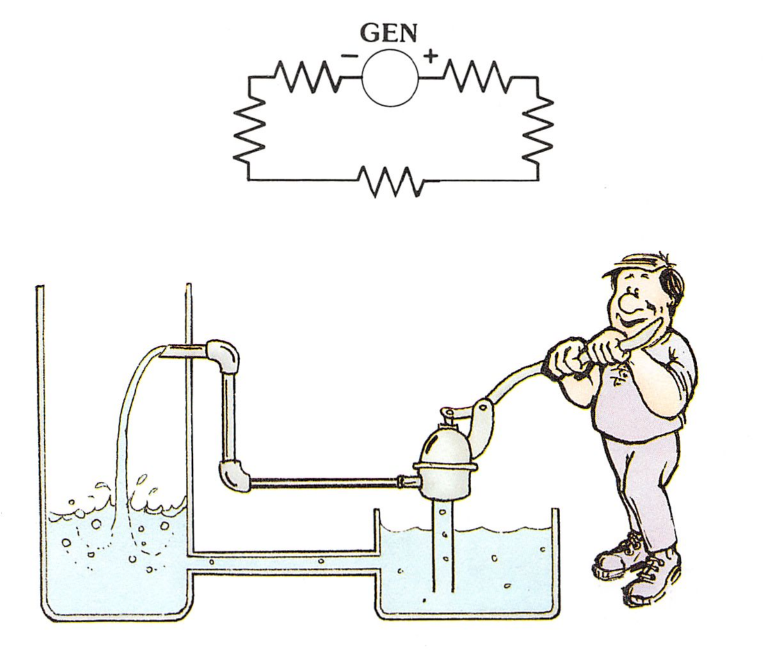
Electricity, like water, is limited in its flow by the size and characteristics of the material it is flowing through – the **conductor**. In a pipe or in a wire, the resistance will remain constant as long as the properties of the conductor do not change. However, the resistance can be altered. In *figure* 7 the side walls of the pipe have been squeezed closer together, thus limiting the passage of the water and increasing the resistance. The same can be done with the flow of electrical current by making use of a variable resistance. When used in connection with electricity, resistance is measured in **ohms** (symbol Ω).

What happens when the pipe is squeezed? If the man at the pump keeps the water at the same depth (and therefore pressure), the flow will be reduced since less water per minute can pass through the neck. The same thing happens with electricity. In this way the voltage, current and resistance are all interrelated. If you change one, it causes one or both of the other two to change.

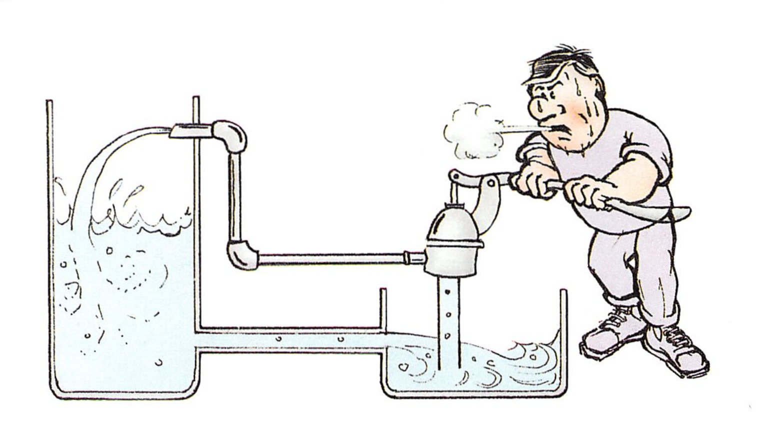
Electricity, like water, must flow to carry information or to perform work. To flow, it must come from somewhere and go somewhere. It is usually convenient to make it flow in a circle to take care of this need, and it is from this that the term electrical **circuit** comes.

In a digital circuit, information is sent by **switching** the current of electricity. This method is used by all modern digital calculators and most computers. On the other hand, in an analogue circuit information is transmitted by varying the current flow. This is the method used in radios, record players and televisions.

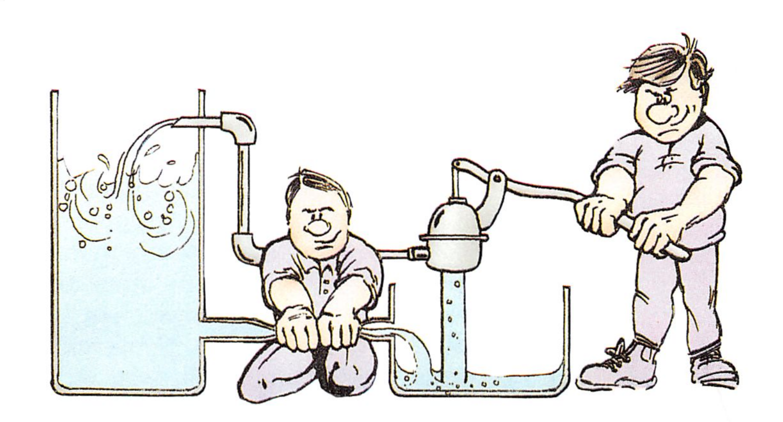
5



6



7



5. **Just like a man pumping water,** electrical voltage from a generator acts to push electrons round the circuit.

6. **The bigger the voltage** the more current flows – the harder he pumps the more water flows.

7. **A resistance in a circuit** is like the squeezed pipe — it reduces the flow of current.

8. **Electrical circuit** of an old fashioned telegraph circuit.

9. **'A' in the Morse code is dot-dash.** A low voltage usually represents 0, a high voltage 1.

10. **A voltage waveform** to represent the binary number 10101.

Sending information by the digital method

Digital computers rely on the same basic method of transmission as a simple telegraphic circuit, although they are obviously far more complex. So by looking at the working of a telegraph code we can get some idea of how this technique might be used in a computer. *Figure 8* represents the diagram of a simple old-fashioned telegraphic circuit. The power supply comes from a battery which pumps electrons to a higher voltage on one side of the circuit than the other. The simple switch in the diagram is the telegrapher's transmitter key. The receiver is a simple buzzer.

In the diagram, the switch is shown as off. Since the voltage is the same on both sides of the buzzer, the receiver is silent. When you press the key, turning the switch on, the voltage on the switch side of the receiver rises, allowing a flow of current and putting the buzzer into action. When the switch returns to the off position, the flow of current stops and the buzzer becomes silent.

Clearly it is the change of voltage in the wire which carries the information. *Figure 9* shows how this works. The level of the bottom horizontal lines represents zero volts, which means that the switch is off (i.e. 'open', as this action opens or breaks the circuit). When the switch is on (i.e. 'closed', as turning a switch on closes or completes the circuit), the voltage rises to the higher level indicated by the upper horizontal lines. If the switch is closed for a short time,you get a Morse Code dot. If it is held closed for a longer period, you get a dash. This waveform indicated gives a dot-dash, which is an 'A' in Morse Code. All this information is provided just by turning the switch on and off.

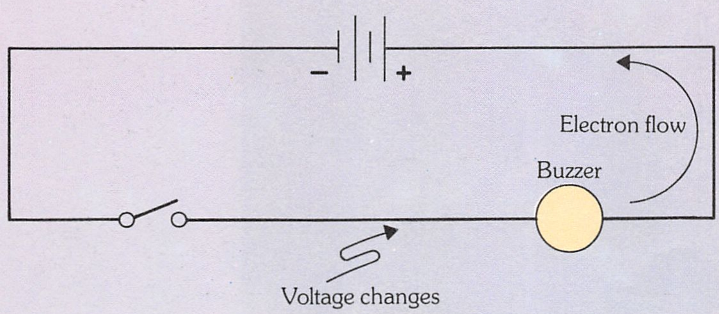
Now look at how this digital method works in a computer. Digital computers are designed to handle information broken down into numerical elements. But numbers in Morse Code are cumbersome, with five characters for every digit (in International Code), so computers use a more efficient code called the 'binary numbers code'. This is how it works. Usually, a low voltage represents zero and a higher voltage number one. But if all that can be transmitted in binary code are zeros and ones, how can the code provide information?

Figure 10 shows a word of five bits; every zero or one is called a bit, and a given number of bits makes up a word. This five bit word will serve as an example, even though typical computers use words of 32 bits. Try reading this word like a number in binary code. The first bit reading from the right stands for one; the second bit for two; the third for four, the fourth for eight; the fifth for sixteen. The zeros are considered as standing for 'no' and the ones for 'yes'. So we can read from right to left in this way:

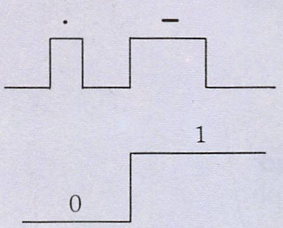
– yes, a one
– no two
– yes, a four
– no eight
– yes, a sixteen.

Adding up the numbers as shown on the bottom line of the figure, we get twenty

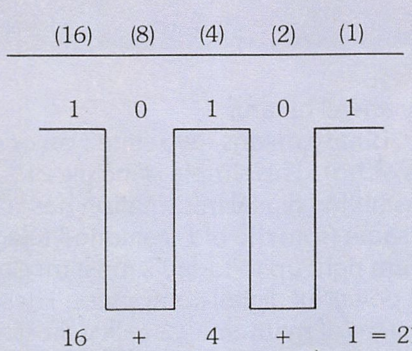
8

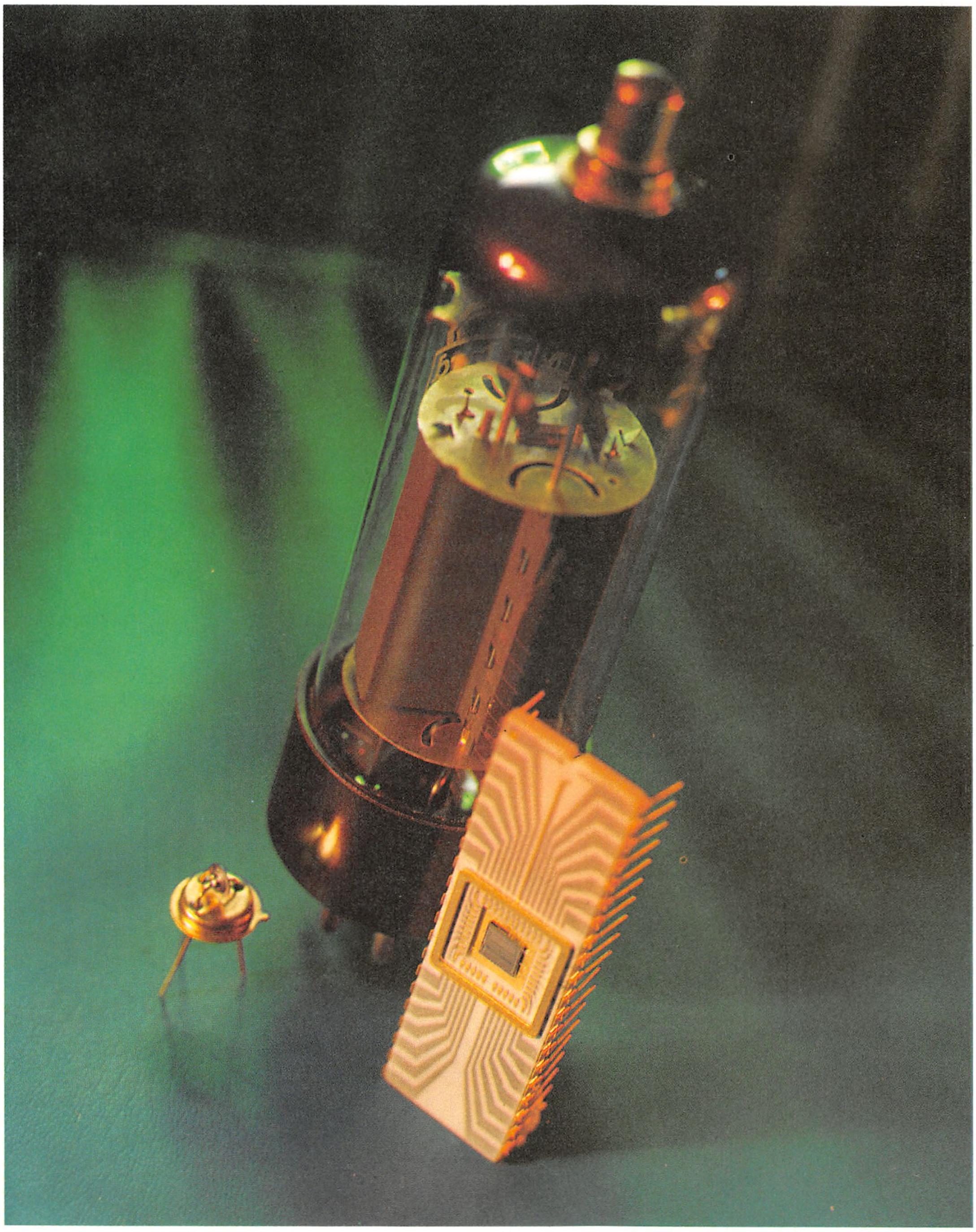


9



10





Paul Brierley

Progress in electronics has been marked by the development of increasingly compact, powerful and reliable semi-conductor devices. The key events were the invention first of the electron tube or valve (the largest component shown), then the transistor, followed by the breakthrough into integrated circuits contained on a single microchip.

one. So the number 21 is represented by the word 10101 in binary code. It's easy to see how more bits can be added on the left. The next bit will represent 32, the one after 64, the one after that 128, and so on. In this way we can transmit numbers of any size and also encode decimal fractions. Digital computers use many other codes such as binary coded decimal (BCD), Gray code and for letters, the Hollerith code. But all these codes use only zeros and ones, so they are all binary.

Binary means 'two state': on or off (yes or no). This simple principle of transmitting digital information has stayed the same from the old-fashioned telegraph system right up to today's most modern and powerful digital computers. These use a system of mathematics called Boolean algebra (which is based on binary counting) to do highly sophisticated calculations.

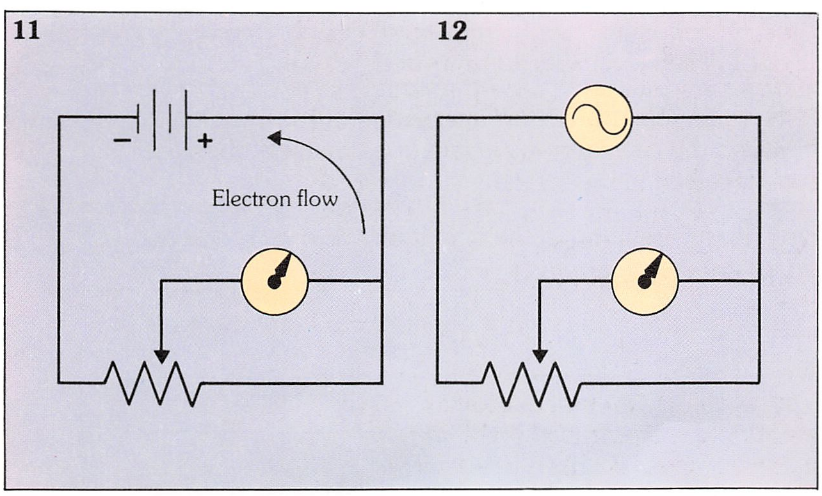
Transmitting information by the analogue method

Digital information controls the flow of electricity by switching it. The only other method available works by varying the current and is called the analogue method. To explain it we can use a very similar circuit to the one used in discussing the digital method. However, in *figure 11* the simple switch is replaced by a **variable resistor** to regulate the voltage and instead of the buzzer there is a meter to measure voltage (this is a special meter with the dial marked in volts). So in this case, the variable resistor regulates the voltage on the line going to the meter.

Right: **An analogue voltmeter** (kindly supplied by G.B.C.).

11. **Circuit for analogue data transmission using direct current.**

12. **Circuit for analogue data transmission using alternating current.**



In the analogue method, a certain quantity of electricity in the line stands directly for the number you want to transmit. If, for example, you let a certain voltage level measurement represent the number, you have a voltage analogue system. Suppose the voltage is adjusted to 10.5 volts by using the variable resistor. On the voltmeter you would then read the actual number 10.5. If you change the voltage through altering the variable resistor, bringing it let's say to 2.5, a different number is read out on the voltmeter.

A large variety of electronic systems use the **voltage analogue method** to transmit information. Most old-fashioned fuel gauges in cars work in this way: a float in the petrol tank controls the variable resistor. As the level of petrol changes, so the voltage going to the fuel gauge alters. This gauge is really a voltmeter whose dial is marked from empty to full instead of in volts. Other examples of voltage analogue devices include analogue computers where the voltage replaces the numbers or the mathematical functions of numbers, and a telephone receiver, which translates varying voltages into fluctuating air pressure, which the ear then interprets as sound.

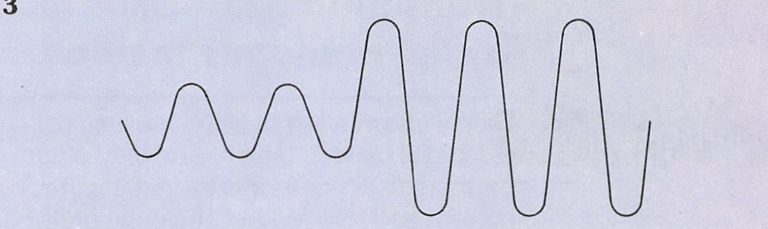
Measurements other than voltage can be used to transmit information. Current analogue systems, for example, work in the same way as the voltage analogue systems except that they depend on measurements of current instead of voltage.

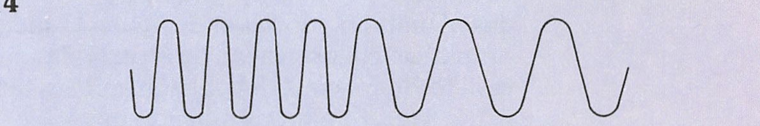
AM and FM

An interesting variety of voltage analogue systems is called analogue amplitude or more commonly **amplitude modulation.** In *figure 12* an alternating current generator has replaced the battery in *figure 11*. Now the voltmeter will oscillate constantly because the generator is producing an alternating voltage, first high with the current going in one direction and then low when the current goes in the other direction. Obviously in this case a constant voltage level can't be established, but the height, or amplitude, of the waves can be measured.

By varying the resistance as in *figure 13* the amplitude of the wave can be changed. In this way we can make the amplitudes stand for the numbers we want to transmit. (AM radios take their name from the fact that they work on the principle of amplitude modulation.)

Another analogue method is that of **frequency modulation.** This technique is used in FM radios. It depends on the waves as does AM but in this case their frequency, rather than their amplitude, is measured. FM waves are illustrated in *figure 14*. Suppose that those on the extreme left, which are close together, represent ten cycles per second (10 Hertz) and the others near the extreme right, which are further apart, have halved their frequency, that is 5 Hertz. So we have managed to represent the numbers 10 and 5. Naturally, there are other analogue methods, but the ones mentioned here are much more commonly used. Briefly, you can say that all analogue methods are based on the regulation of various properties of electricity, while all digital methods are based on the switching of electricity.





13. **Amplitude modulated (AM) waveform.**

14. **Frequency modulated (FM) waveform.**

Glossary

amplitude modulation	method of transmitting information in the circuit by varying (modulating) the voltage of the electrical waves
analogue	method of transmitting information in an electric circuit by means of varying the current or the voltage
current	flow of electrons, usually expressed in Amperes (A)
alternating current (a.c.)	electrical current in which the flow of electrons changes direction at regular intervals
direct current (d.c.)	flow of electrons in one direction only
digital	method of transmitting information in an electric circuit by means of switching the current (on-off)
electrons	the minute particles of which electricity is composed
frequency	the number of times per second that an alternating current makes a complete cycle through the circuit (going first in one direction, then the other). Nowadays expressed in Hertz (Hz)
frequency modulation	method of transmitting information in the circuit by varying (modulating) the frequency of the electrical waves
resistance	the difficulty of passing current through a conductor when voltage is applied. Usually expressed in ohms (Ω)
voltage	pressure of the electrons, usually expressed in volts (V)

1

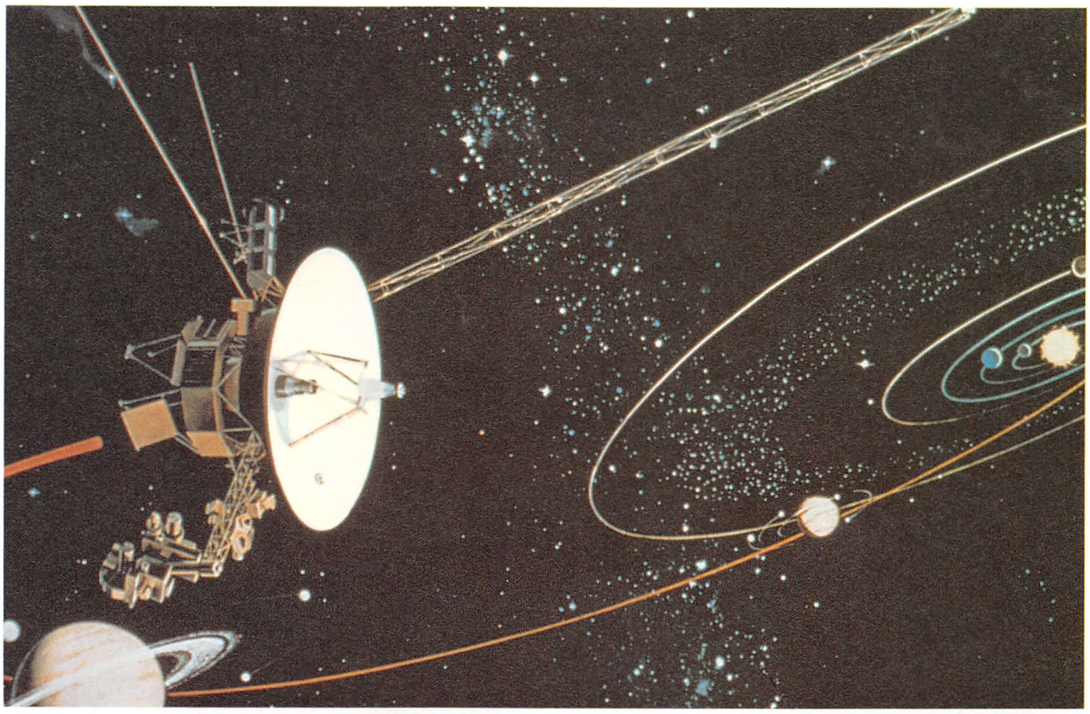
DIGITAL ELECTRONICS

Basics of a digital system

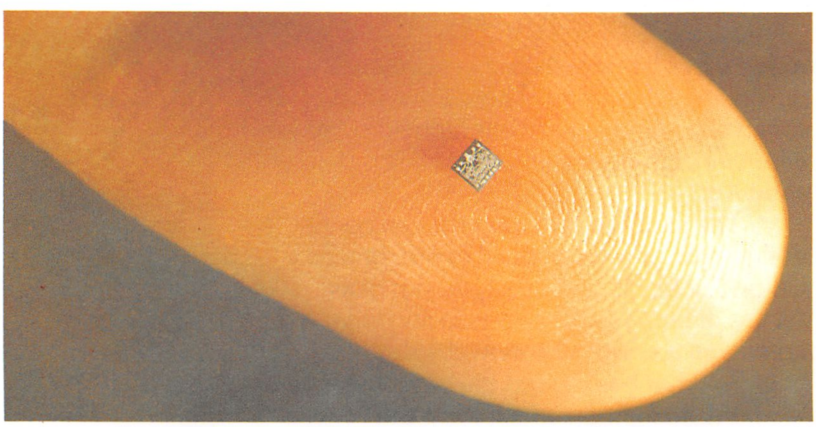
The digital take-over

Recent years have seen the birth and development of a new generation of electronic devices. Pocket calculators, computers and video games have now taken their place in our everyday lives, alongside the radio, television and hi-fi. What sets apart this new generation is the extensive use of digital control systems.

In all electronic devices the information to be processed, controlled or transmitted is in the form of voltage or current signals. However a quick comparison between, for example, a radio and a pocket calculator demonstrates an important difference in the kind of signals used. One type (the radio) is described as linear or **analogue,** and the other (the calculator) is known as **digital.**



Today's most sophisticated technology relies on digital electronics. **Below:** a single chip integrated microcomputer.



Tony Stone Photo Library – London

An analogue signal, such as that used in a radio, is one that changes continuously. It is the fluctuating size of the signal which is measured and processed by the analogue device. Signals like this, which can be of any value within a given range, cannot be stored in an electronic memory. The precision with which analogue signals can be processed is thus limited.

Digital systems on the other hand process signals which can have just two

logical values: 1 or 0; true or false. These signals are easily represented by two voltage levels (e.g. low for 0, high for 1) and as a result digital circuitry is relatively simple, reliable and immune to interference. Information in this form is also easier to store in an electronic memory.

Digital and analogue circuits are similar in as much as they are both constructed using the same type of circuit elements – diodes, transistors, resistors etc. – to transfer electrical signals. So simply looking at the circuitry cannot always reveal if it is digital or analogue. The difference lies in the digital processing of the signals; this is what makes the system such a powerful tool — able to perform incredibly complicated tasks yet neither large nor expensive.

The name digital comes from the word 'digit'. A digit is any single number from 0 to 9. Ultimately all the information fed into the digital system is translated into a code which consists of the simplest possible numerical elements. These are capable of assuming just two values, 1 and 0.

It seems that every day new ways are found to apply digital systems, sometimes to replace analogue devices but also in totally new applications, handling jobs that were once considered impossible.

To keep up with the rapid progress in the field we need to become familiar with a whole new language. As well as the hundred and one terms involved in analogue electronics – such as microphones, loudspeakers, transformers, amplifiers, oscillators, synthesizers – there are now new words like gate, flip-flop, counter, register, decoder, binary number, TTL, MOS, microprocessor.

Digital electronics is where the future lies, and these chapters will explain the factors common to all digital systems, looking at digital electronics at work in a vast range of applications, from the light switch in your own home (did you know this is a simple digital device?) up to the biggest computer.

You don't have to be a specialist to understand in some detail why digital methods are revolutionizing the electronics field, and to see what sorts of things can be expected in the future from this amazing technology.

A simple calculator as an example of a digital system

To show exactly what a digital system is and how it works, let's start with a piece of equipment that, in all probability, you are already familiar with, a small calculator such as the simple model shown in *figure 1.* It's really a very economical device, bearing in mind the amazing things it can do. Just a few years ago, an electronic calculator capable of adding, subtracting, multiplying and dividing was as big as a typewriter and cost as much as a small car – yet another example of how digital systems are becoming ever more sophisticated, small and cheap.

Have a look now at what the calculator can do, examining closely the sequence of events. Switch it on. Press key 3 watching what happens. Look a little more closely at the number itself and how it is displayed. If your calculator is of the most common type, you will see that the 3 is made up of five small distinct segments, which are illuminated. There are a total of seven segments in each character display – when all of them are lit a figure 8 is showing.

The illuminated segments can consist of fine bars or rows of dots in red, green or yellow, or black bars which do not glow. All are different ways of producing what's known as **a seven segment display.**

So what has caused the particular five segments to light up which form the

1. An entire digital system you can hold in your hand! The first electronic pocket calculator from Texas Instruments.

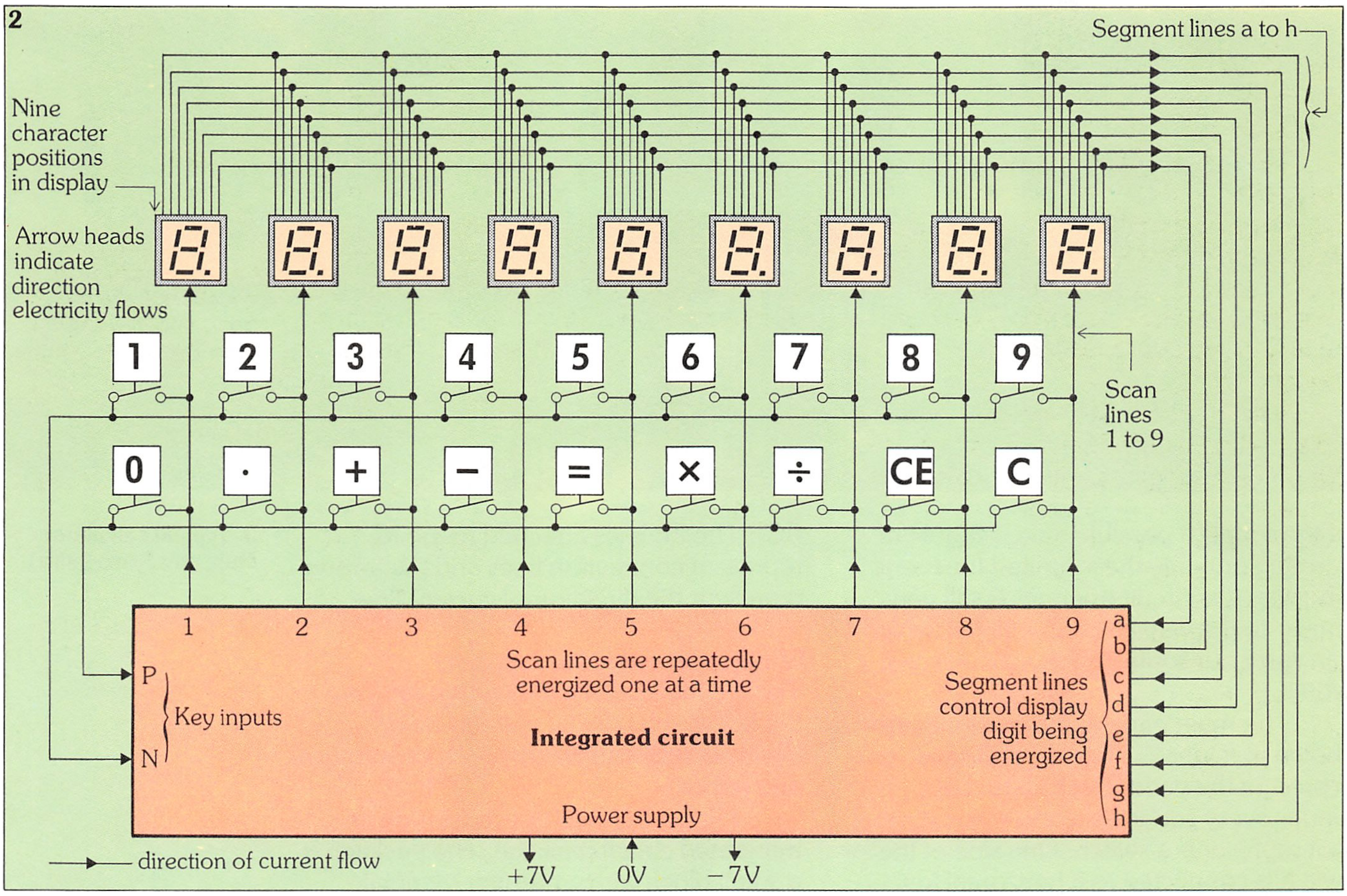
number 3? Pressing key 3 has sent some sort of information to an internal part of the calculator; information which means 'Take note of the number 3'. And in some part of the internal circuitry this memorized number 3 has caused the correct five segments to be illuminated.

Try now to follow the necessary operations to add 5 to 3 and obtain the result. The particular keys to be pressed, and the order in which they are pressed, depend on the type of calculator you are using. Probably you will have to press

Where has the 3 gone? It has been memorized somewhere, even if it is no longer displayed. When you press the '=' key after the 5 something causes the numbers to be added together, the 5 disappears and the correct result appears on the display, i.e. 8. But how has the calculator carried out the calculation? And what has happened to the 3 and the 5?

Just to add two numbers, both of which are less than 10, a number of complicated things have happened inside this little machine. Understanding how the

2. Schematic diagram of connections between IC chip, keyboard, and display in a simple calculator.



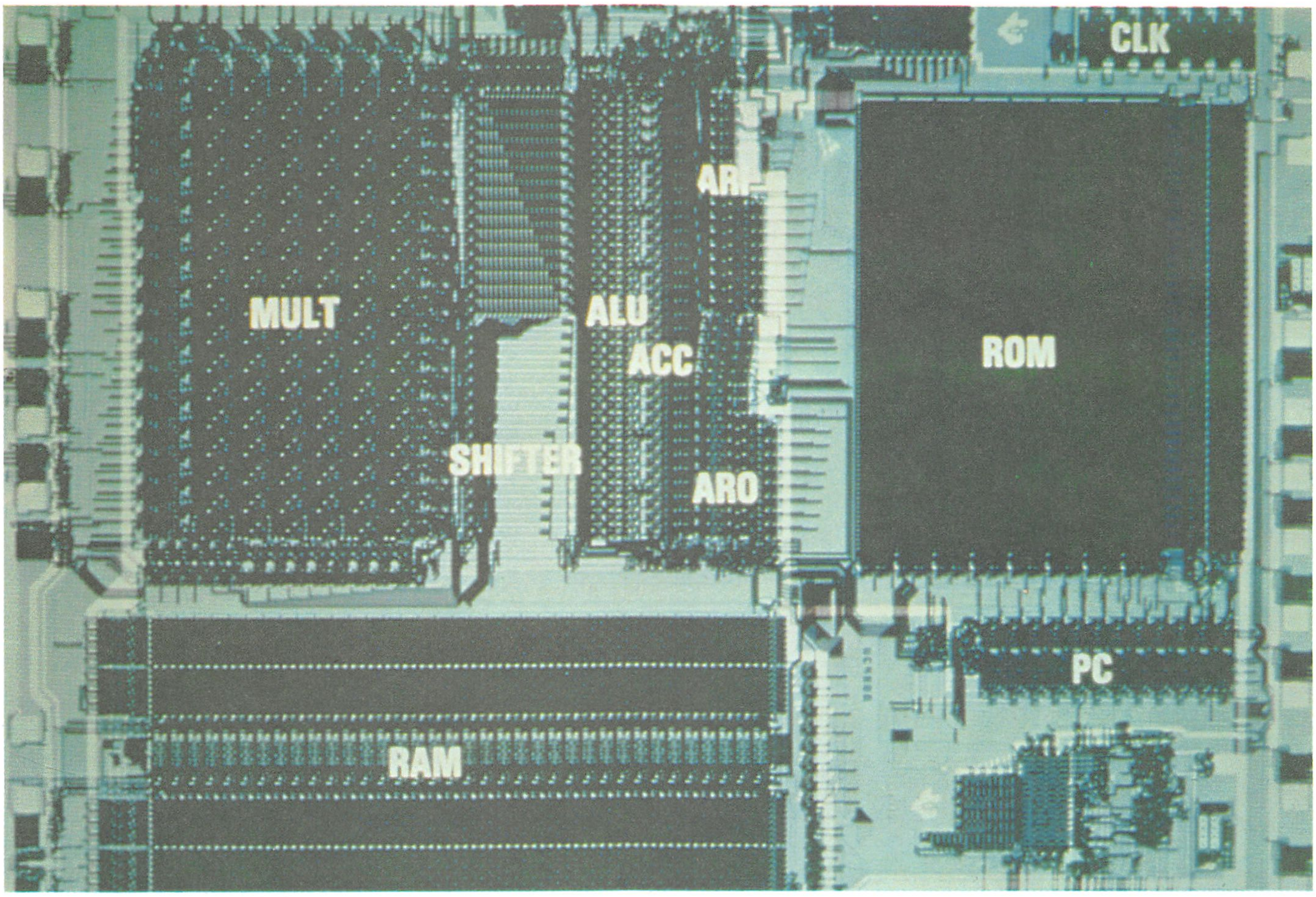
sequentially the '+' key, then the number 5, and finally the '=' key. Watch what happens while you are pressing the keys to get the sum of 3 plus 5.

When you press the '+' sign the 3 remains and, depending on your type of calculator, maybe a '+' sign is displayed. Having pressed the '+' key something inside the calculator has taken note that the next number entered has to be added to the one entered previously. When the 5 key is pressed 5 takes the place of the 3.

numbers are relayed from the keyboard, how they are added together, memorized and displayed, provides a good outline of the basic workings of a digital system.

Starting with a simplified calculator

For the purposes of this explanation, imagine a very simple calculator which is capable only of addition, subtraction, multiplication and division, and can display a number up to eight digits long. The electronic circuitry of this imaginary calculator is



3. Typical calculator chip, greatly magnified.

of the simplest possible type, capable of carrying out only these limited functions. The way this circuit operates is still very similar to the way a real calculator, which can carry out sophisticated calculations, works.

The main parts of the machine are shown in *figure 2*. The 18 small squares represent the various calculator keys. Under every square is a schematic symbol that represents a switch. One side of the switch is connected to a horizontal **key input** line, marked N or P. The other side is connected to a vertical **scan line** (numbered from 1 to 9). The 'lines' are in fact metal tracks on a printed circuit board.

Above the key squares on the diagram, are shown some slightly larger boxes with the number 8 inside. These represent the display which can show numbers up to eight digits long as well as the '–' sign for negative numbers, and other symbols (such as an E) to indicate error.

The big block at the bottom of the diagram represents an **integrated circuit** (IC). The 22 lines attached to the IC represent connection lines and the arrows represent the direction of current flow.

What is an integrated circuit?

The technology of circuit integration is the key to why digital systems have become increasingly compact, economical and refined.

Figure 3 shows the chip inside an integrated circuit package. The package is a small block with a number of metal pins which form the electrical terminals. These pins are connected inside the IC to a small piece of semiconductor material (a chip) made of silicon, with dimensions of 6 × 6 mm and a thickness not much more than a page of this book. The dark areas are hundreds of transistors, diodes, capacitors and other circuit elements crammed together so closely it is impossible to distinguish between them – even when the circuit is magnified many times. Most small calculators have all their electronic circuitry squeezed onto a single chip and the rest of

the machine is taken up by the keyboard, battery and display.

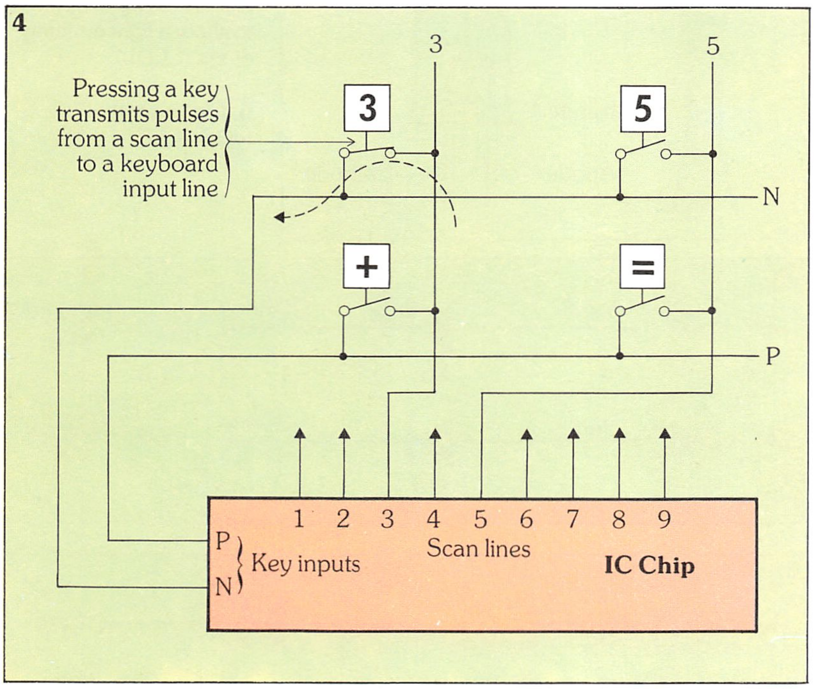
Just how this integrated technology is created will be covered in detail in a later chapter.

To help understand how the keys transmit numbers and commands to the IC, an enlarged section of the keyboard is shown in *figure 4*. The IC applies a voltage to each of the vertical scan lines – in fact the voltage is being applied sequentially to each of the lines thousands of times per second. When a key is pressed (e.g. key 3) the switch below it closes. This generates a connection between scan line 3 and input line N, so that the voltage pulses being applied to scan line 3 appear on line N.

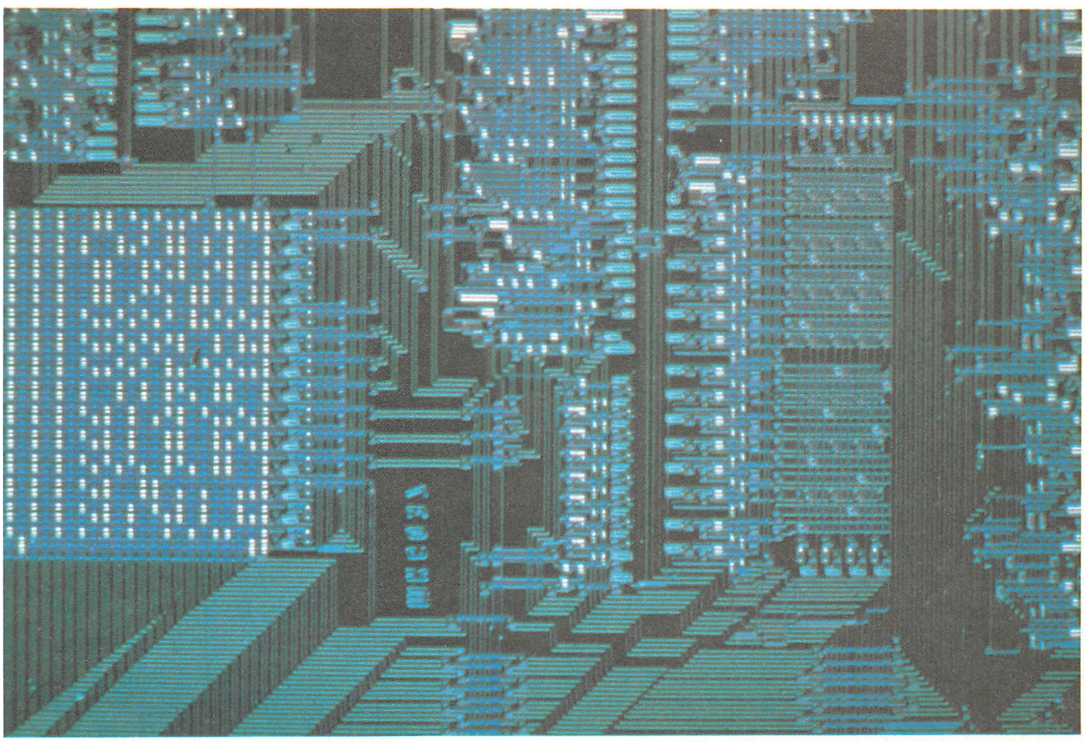
Unless it is busy calculating, the IC constantly monitors lines N and P, checking for a signal. During even the briefest depression of the key, the IC will have checked all the scan lines many times. Every time that scan line 3 has a pulse applied to it while the key is depressed input line N receives a voltage pulse. It doesn't matter how fast the key is pressed – the speed of the digital circuit compared to the fastest mechanical switch is like comparing a racing car to a tortoise.

The IC knows that each scan line sends pulses to just two keys, so by looking at which of the input lines is pulsing it can tell exactly which key is pressed. For example when input line P, rather than N, is energized when scan line 3 is pulsed, it indicates that the '+' key has been pressed rather than key 3. When key 5 is pressed and scan line 5 is pulsed, input N receives pulses. Looking at *figure 4* makes it quite easy to follow through these connections.

4. Detail of fig. 2 indicating the pulse flowing from scan line 3 into input line N when calculator Key 3 is depressed.



Right: microphotography reveals the complexity of an integrated circuit.



LED displays

Having seen how the numbers arrive at the IC from the keyboard, let's now look at how the numbers are displayed. We'll start by looking briefly at how LED displays work (this will be treated in more detail in a later section).

Figure 5 shows the symbol for an LED, **Light Emitting Diode**. An LED is a diode (a device that allows current to flow in one direction only) that under certain conditions emits light. In general it is a low cost device and many different types and sizes are available. LEDs come in a range of different colours – reds, greens and yellows are the most common – although the red ones are the cheapest and most popular. In addition there are diodes which emit light in the infra-red band for use in remote controls and alarms.

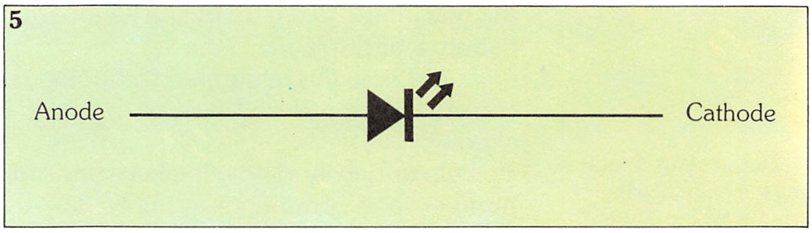
A typical red LED consists of a small semiconductor chip mounted on, and in electrical contact with, a metal base. A fine wire is connected to the upper surface of the chip (see *figure 6*). The assembly is covered with epoxy resin which, in effect, acts as a lens to magnify the emitted light. This magnification is in the order of 2.5 times.

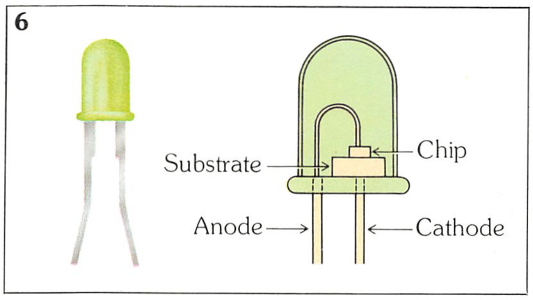
All diodes have two terminals, called the anode and the cathode. When a voltage is applied to the anode, making it more positive than the cathode (a condition known as forward bias) then current will flow through it. If the cathode is made more positive than the anode (reverse bias) then the diode will not allow current to pass. This happens because the resistance of a diode is much lower under forward bias than its resistance under reverse bias.

A light-emitting diode emits light when current is passing through it, i.e. when it is forward biased.

It is important, however, that the current flowing is not too high, otherwise the diode may be damaged, nor too low in which case insufficient light will be emitted.

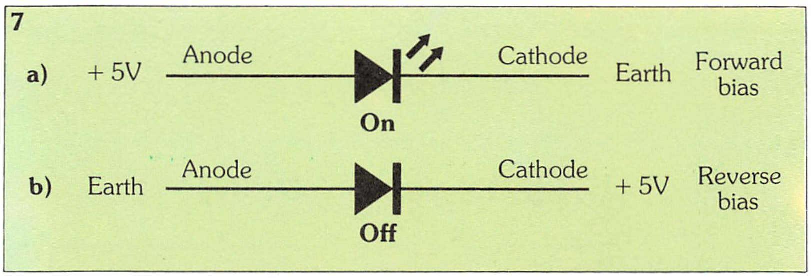
The typical current required by an LED to operate safely is in the range 5 – 40 mA (milliamps, or thousandths of an amp) with a voltage drop across it of 1.8 V. If a voltage of 5 is used in your circuit this means that there are 3.2 V too many for

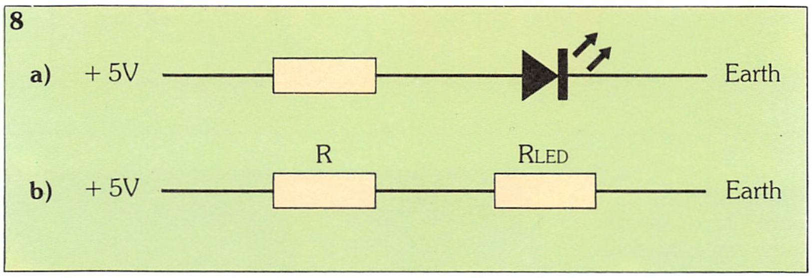




5. Usual symbol used in electric circuit diagrams to show a light emitting diode (LED).

6. Cross section of a typical LED.





your diode. To stop the extra voltage from damaging the diode a resistor is included in the circuit. The value of the resistor is calculated to reduce the voltage across the diode to the required value. This is shown in *figure 8*.

Figure 9 shows a circuit permitting an LED to work as a logic state indicator, i.e. to light up only when voltage is applied to it in the forward bias direction. If a voltage of 5 V (logic 1) is applied *(figure 9b)* the LED lights up but a zero voltage (logic 0) causes it to switch off *(figure 9c)*.

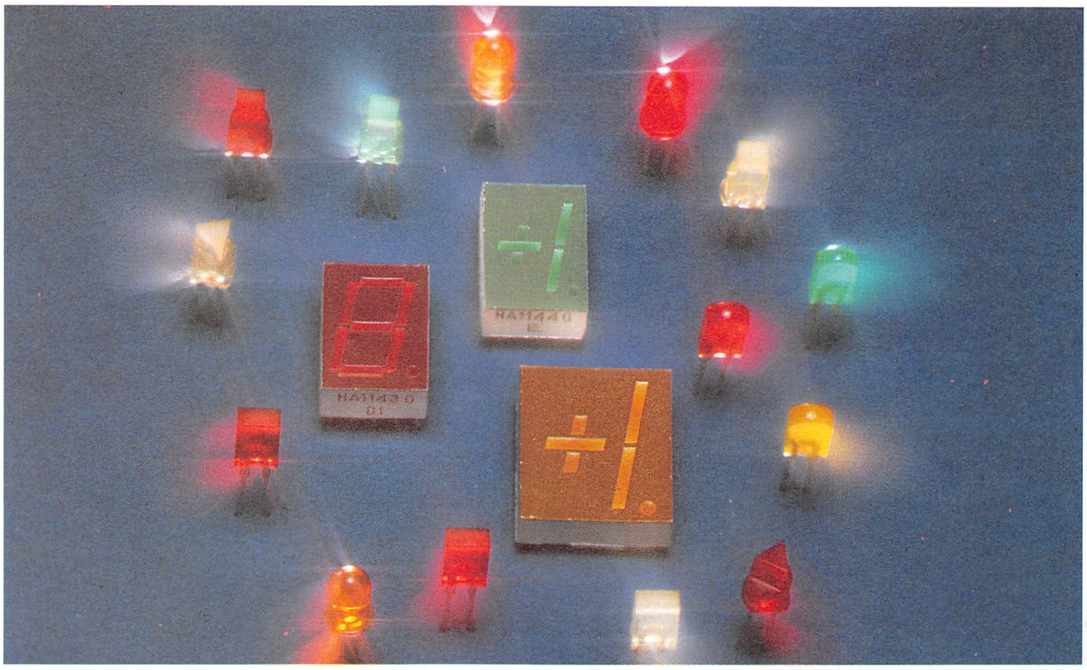
This circuit is perfectly acceptable as a logic state indicator. However, there is a problem with it – it needs more current (10 mA) when switched on than can be delivered by an integrated circuit.

Figure 10 shows a way round this.

7. How an LED works. Current flows and the LED emits light only when the anode is at a higher potential than the cathode.

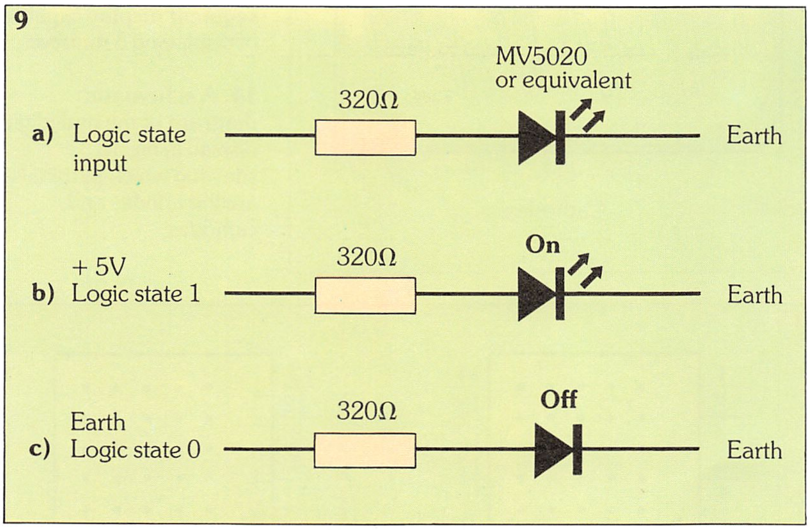
8. a) current limiting resistor and LED
b) the equivalent electrical circuit

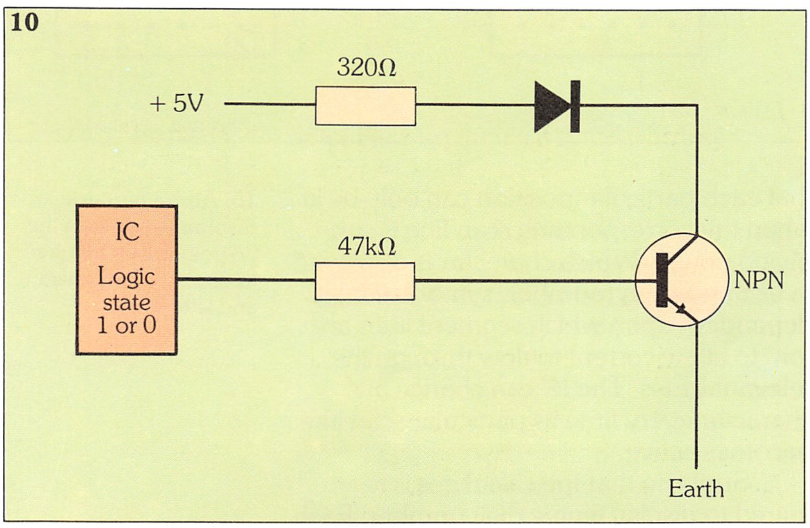
Right: three types of display surrounded by LEDs of various colours (kindly supplied by Siemens).



9. Use of the LED as an indicator of logic state, with current limiting resistor.
a) the circuit
b) logic state 1, LED lit
c) logic state 0, LED off

10. Improved LED logic state indicator requiring a much smaller current from an integrated circuit by using a transistor.





The circular unit in the centre represents an NPN transistor. This device, together with a large (47 kilohm) resistor, effectively can switch on the LED when it receives just a tiny current (0.9 mA) from the IC. The workings of a transistor amplifier are explained in detail in a later chapter.

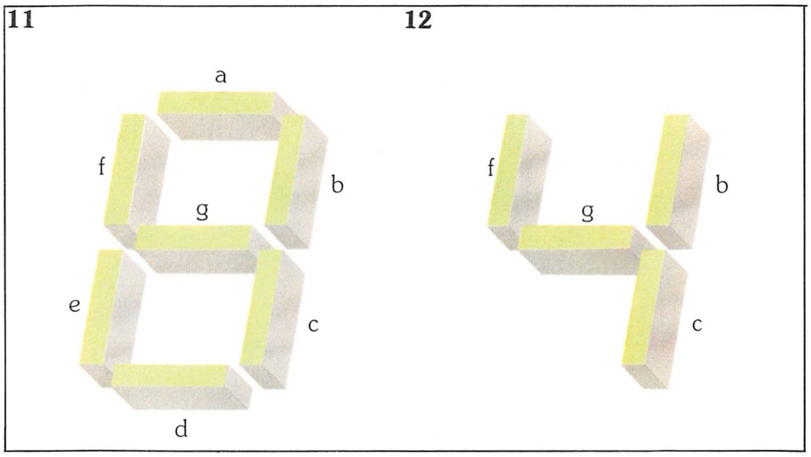
We usually want to display numbers and letters rather than just indicate a logic state. A seven segment LED display can be used to display numbers (see *figure 11*). By switching on the appropriate segments, numbers between 0 and 9 can be displayed. For example to display the number 4, segments **f, g, c** and **b** would be lit *(figure 12)*. To display multi-digit numbers an arrangement of the type illustrated in *figures 13* and *14* is used. This is a five-digit example.

Eight anodes and five cathodes may be used to control a total of 40 segments i.e. 35 segments which make up the five numbers plus five decimal points. Any one of the LEDs can be lit by selecting one of the eight anodes and one of the five cathodes. In effect a display of five digits each containing seven segments is controlled through an 8 × 5 matrix. Just as for the single digit display in *figure 11*, a segment is lit when the cathode is earthed and the anode is connected to 5 volts via a current limiting resistor.

To display letters as well as numbers,

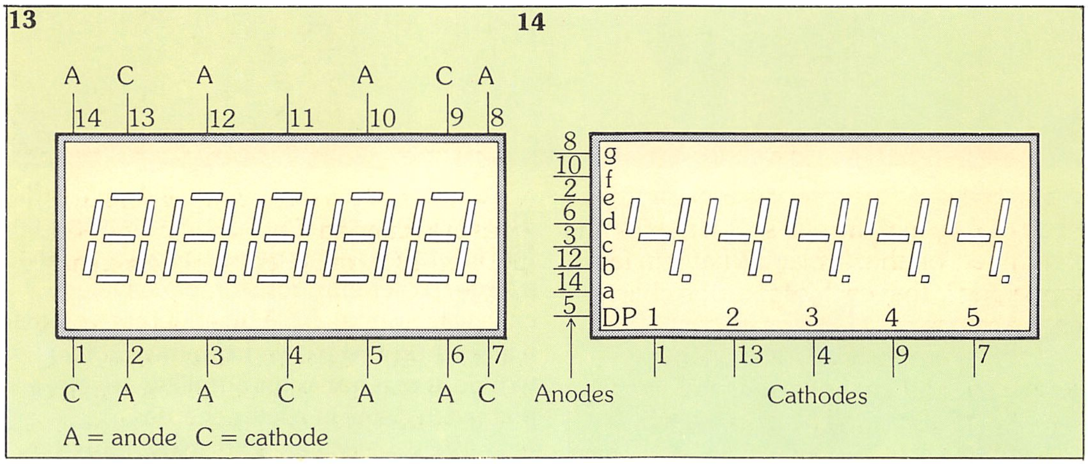
an alphanumeric display is used *(figure 15)*. This consists of 36 dot-sized LEDs arranged in a seven line five column matrix. These display a number or letter and a separate LED is used for the decimal point. This is mounted in a plastic package with a transparent epoxy window, with seven connection pins on two of its sides. A fairly sophisticated IC is required to drive an alphanumeric display. There is no need to understand this in detail at this stage, but the basic principle used to generate a character is as follows.

Take the display of letter 'T' on the 7 × 5 point matrix as an example. Assuming that the points to be illuminated corresponding to the letter are to be in logic state 1 and the background is to be in logic state 0, the logic state matrix shown in *figure 16* is required. This in turn gives a series of illuminated points to form the letter 'T' as is shown in *figure 17*. The IC controlling the display ensures that only the points required are illuminated.



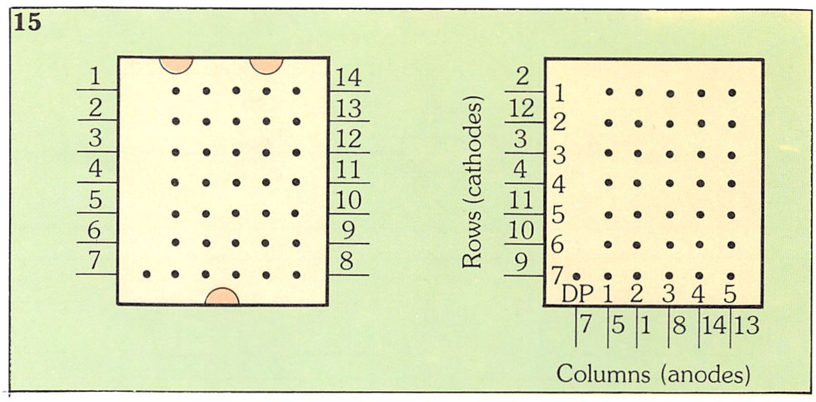
11. A seven segment display.

12. Segments illuminated to display the number 4.



13. A multi digit seven segment display capable of displaying 5 numbers.

14. A schematic diagram of the multidigit display in fig. 13 showing which terminals are the anodes and cathodes.



15. Alpha numeric display — showing the 36 point LEDs required to display all numbers and letters.

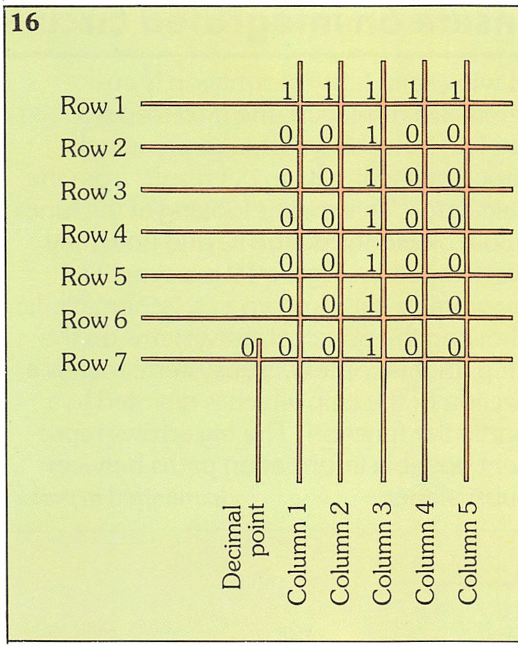
Look again at *figure 2* in the light of what you now know. Each of the top squares represents a digit, the minus sign or the error symbol; it is also possible to display a decimal point to the right of each digit. Each of the nine positions is connected to a vertical scan line and to the eight segment lines **a** to **h** ('h' controlling the decimal point). All of the eight segment lines are connected to each of the nine characters and to the IC.

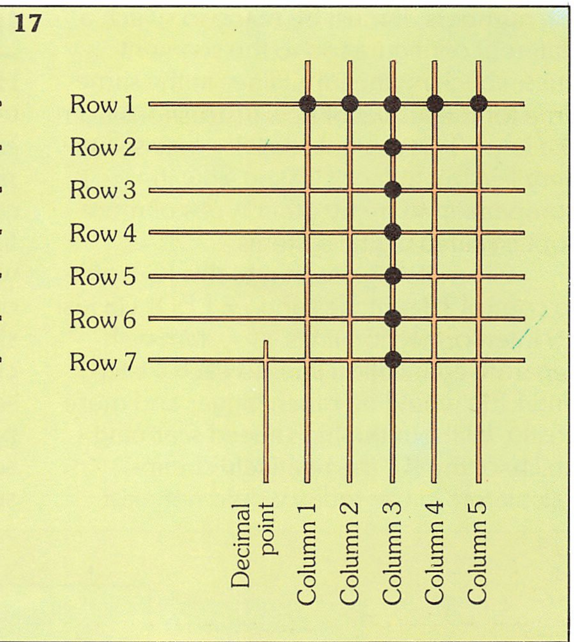
Figure 18 shows the connections to one of the individual characters illustrated in *figure 2*. The circuit configuration allows the IC to control the display in such a way that each particular position can only be lit when the corresponding scan line is supplying power. Which character is displayed in each position (number, symbol, etc.) is dependent upon which segment lines are 'on' to allow current to flow through the relevant LEDs. The IC can change a character every time its particular scan line becomes active.

Suppose that the calculator is required to display a nine digit number. Each

16. The logic state of the LEDs in an alpha numeric display required to show a letter T.

17. How the T appears.



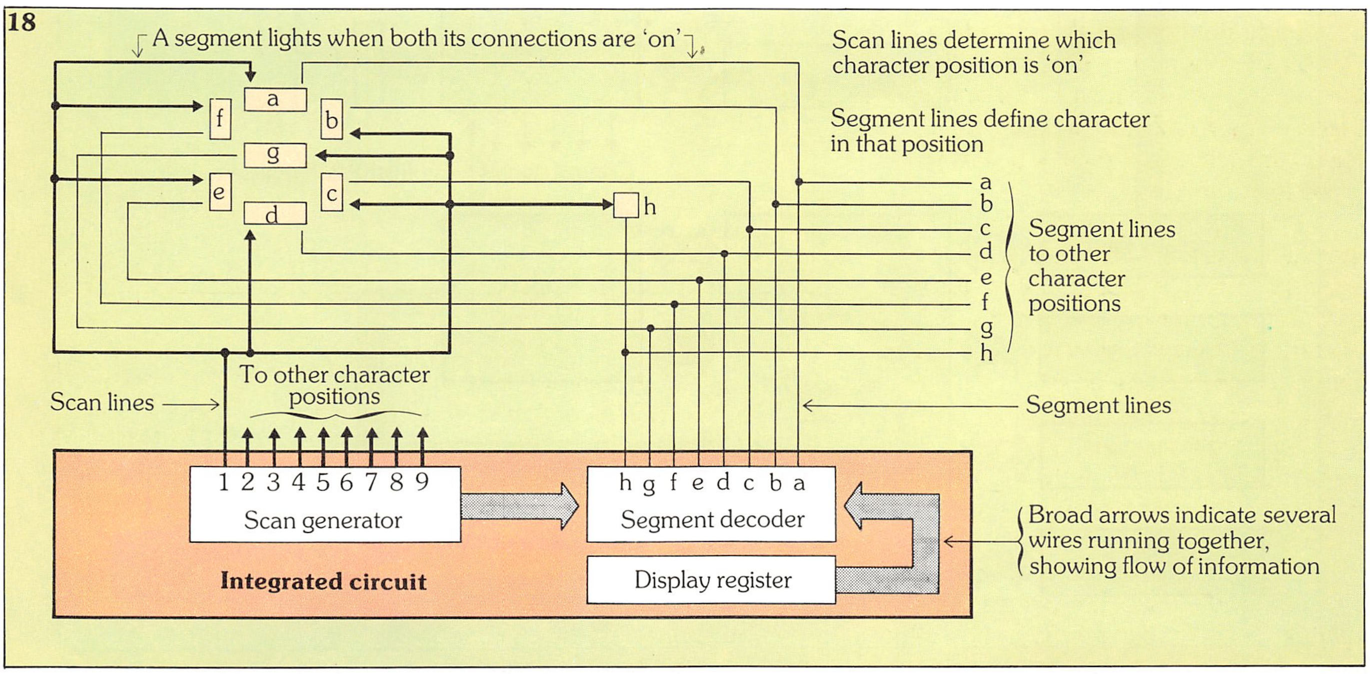


of the nine characters is switched on and off so rapidly that the eye sees a fixed string of numbers on the display. What is in fact happening is that each of the nine characters is activated in turn.

For example, when line 9 is active along with segment lines **a, b, c, d, g** and **h,** (see *Fig 11*) a 3 will be displayed in the extreme right position with a decimal point after it. As normal scanning continues, and line 9 deactivates, scan line 1 is turned on and the 3 and its associated decimal point is momentarily switched off whilst the other lines are scanned. Each character from position 1 (extreme left) to 9 is illuminated in turn.

The rate at which the characters are switched on and off is so fast that even if each character is only on for one-ninth of the total period of each scan, the eye sees a fixed string of characters on the display. As you can well imagine, the IC is carrying out a tremendous amount of work, even if it is not doing calculations but simply display-

18. Schematic diagram showing details of connections to eight LED segments at the far left character position in the calculator display shown in figure 2. Arrowheads show direction electricity flows.



ing numbers. It must be ready to switch a different configuration to the segment lines, checking the input lines at the same time for the arrival of new instructions from the keys. This multiple activity may seem very fast but in fact it is very slow in comparison to many other types of more sophisticated digital systems.

Given its high scan rate this IC is able to control 18 switches and 72 LEDs via just 19 interconnection lines. A system with separate connection lines to each switch and LED would be much bigger and more costly. Without the high speed scanning ability of the IC, the relatively cheap calculators in use today would not exist.

Inside an integrated circuit

Having seen how information is entered and displayed, the next step is to explain how this information is actually processed in the IC to get results from the calculator. This means looking at the functional blocks inside the IC and how they work together. *Figure 19* is a simplified diagram (what's known as a 'schematic'), showing the principal subsystems on the chip. In this context, 'subsystem' means a section of the chip which is devoted to a particular function. The big arrows represent possible information paths between subsystems. (*continued in part 2*)

19. Simplified diagram of the major subsystems in a simple calculator, showing the pathways between the subsystems.